Research Article

A Multidata Connection Scheme for Big Data High-Dimension Using the Data Connection Coefficient

Yoon-su Jeong¹ and Seung-soo Shin²

¹*Division of Information and Communication Convergence Engineering, Mokwon University, 88 Doanbuk-ro, Seo-gu, Daejeon 302-318, Republic of Korea*

²*Division of Information Security, Tongmyong University, 428 Sinseonno, Nam-gu, Busan 608-711, Republic of Korea*

Correspondence should be addressed to Seung-soo Shin; shinss@tu.ac.kr

Received 23 November 2014; Revised 11 March 2015; Accepted 12 March 2015

Academic Editor: Bo Yu

Copyright © 2015 Y.-s. Jeong and S.-s. Shin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the era of big data and cloud computing, sources and types of data vary, and the volume and flow of data are massive and continuous. With the widespread use of mobile devices and the Internet, huge volumes of data distributed over heterogeneous networks move forward and backward across networks. In order to meet the demands of big data service providers and customers, efficient technologies for processing and transferring big data over networks are needed. This paper proposes a multidata connection (MDC) scheme that decreases the amount of power and time necessary for information to be communicated between the content server and the mobile users (i.e., the content consumers who are moving freely across different networks while using their mobile devices). MDC scheme is an approach to data validation that requires the presentation of two or more pieces of data in a heterogeneous environment of big data. The MDC transmits the difference of the consecutive data sequences instead of sending the data itself to the receiver, thus increasing transmission throughput and speed.

1. Introduction

The increase in data sources, including social networks and mobile devices, has brought the exponential growth, availability, and use of information [1, 2]. This trend to larger data sets, commonly referred to as big data, entails new technologies and practices of collecting, storing, and analyzing data sets as traditional data processing applications cannot handle them efficiently. Advanced big data technologies can uncover hidden values or additional information from large volumes of data, providing more predictive, tailored information that can lead to substantial innovation gains [3, 4].

Although the immense amount of data is generated on a regular basis via a variety of channels (e.g., business processes, machines, and human interaction with things like social media sites, and mobile devices), information services reflecting specific individual and community needs and interests are scarce. This indicates that technologies that can uncover the insights and meaning of the underlying data are essential to gain value from big data.

Big data implies enormous volumes of data. For example, a single data file in scientific applications is of terabyte size and it can take hours, days, or even longer to traverse the network. However, volume is not the only aspect that characterizes big data. Big data is often defined along three dimensions—volume, velocity, and variety [5]. That is, big data is high volume, high velocity, and high variety information assets.

Big data has the power to radically change every aspect of our lives, including politics, economy, culture, and science. While it provides new opportunities for prosperity and innovation, there is always the question of privacy or security, particularly with customer data [6]. Online searches, store purchases, Facebook posts, cell phone usage, and so forth are creating a flood of data that when organized and analyzed, reveals trends and habits about individuals and society at large. This is why big data is sometimes nicknamed “modern day version of Big Brother.” The threats of data breach and leakage need to be understood and protected through the use of appropriate security controls.

TABLE 1: Four elements of big data.

Element	Descriptions
Volume increases	(i) Digital information is growing exponentially along with the advance of IT (ii) Enter the zettabyte era
Variety increases	(i) Data collection can be from varied sources and forms (e.g., activity logs, text posted on social media, spatial data, and physical event data) (ii) The diversity of unstructured data grows (e.g., multimedia data)
Complexity increases	(i) Data that does not fit the structures of formal database architectures, incompatible data storage formats, redundancy, and so forth (ii) The complexity of data management grows due to the diversity of data types, forms, and sources (iii) Advanced techniques are required to efficiently manage and process the data of high complexity
Velocity increases	(i) The increase of real-time data (object data, sensor monitoring data, streaming data, etc.) (ii) The speed of data generations and flows are fast-paced (iii) Processing and analysis speed is vital to deal with huge volumes of data and to provide volatile information in time

Big data is being transmitted over communications networks with increasing frequency, causing the problem of machine and communication overhead. The transmission of huge data files is not the only challenge. Real-time data transmission is also becoming increasingly commonplace. For example, mobile users may request big data services using their smartphones while moving across different networks. An efficient technique that can process and move large amounts of data quickly and easily in such a setting is needed.

The MDC scheme proposed in this paper minimizes the time necessary for information to be communicated between the content server and the mobile user so that the user moving across networks can receive big data services without interruption. MDC scheme is an approach to data validation that requires the presentation of two or more pieces of data in a heterogeneous environment of big data. The MDC transmits the difference between the consecutive data sequences (the locations of the data in the content server, more precisely) rather than the data itself. This allows achieving low latency and reliable throughput, thus avoiding interruptions in service. In addition, the MDC delivers sufficient random bits to be used for validation checks.

The remainder of the paper is organized as follows. Section 2 describes the definition and characteristics of big data. Section 3 presents the proposed MDC scheme. In Section 4, the performance of the proposed scheme is evaluated. The final section provides concluding remarks and directions for future research.

2. Related Work

2.1. Big Data. Big data refers to the expanding volume of high velocity, complex, and diverse types of data, both structured and unstructured [1]. Big data comes in the form of emails, photos, monitoring devices, audio, cell phone GPS signals, and many more. With the widespread use of mobile devices and the Internet, data is routinely generated by human interaction on systems like social media, causing

the exponential growth of the volume of data to be analyzed. The advance of machine to machine (M2M) also accelerates the exponential growth of data volumes [7, 8].

In today's world, data comes from everywhere and it is different from the data in conventional database systems in many aspects. Online content produced by Internet users, commonly known as the UCC or UGC, is an example of the unstructured data that creates problems for storage, mining, and analyzing data. With the available data on social media platforms, it is possible to know many personal things about individuals (e.g., demographic information, preferences, purchase habits, and friends). CCTVs that are installed almost everywhere (e.g., roads, buildings, and inside the residence elevator) are gathering huge volumes of video data every day. Along with the private sector, the public sector is generating and managing large amounts of data associated with census, health and welfare services, national pensions, and so forth [9].

2.2. Characteristics of Big Data. Table 1 shows the four major elements of big data—volume, velocity, variety, and complexity. Today, big data is recognized as a government and national asset in many countries but it requires innovative forms of information processing for enhanced insight discovery, decision making, and process optimization. The size of the data sets within the data analysis and velocity with which they need to be analyzed has outpaced the current abilities of standard tools and methods of analysis.

Technologies such as distributed storage, parallel processing, and cloud computing allow fast processing and analysis of massive data sets. With such technologies, industries and digital marketers can gather immediate feedback from tweet updates, Facebook comments, or social media discussions and predict and analyze their business circumstances within a tolerable elapsed time. Thus, companies can provide a more appropriate and prompt response to the customers, increasing their chance of success.

In the market, there are open source and/or free software solutions and distributed, scalable, and flexible hardware infrastructures for the management and analysis of large quantities of data. For example, Hadoop, R packages, and cloud computing are popular big data technologies. That is, it is possible to realize benefits from the use of big data without procuring expensive storages or a data warehouse [4, 10].

3. Multidata Connection Scheme

As the volumes and types of data in clouds increase, techniques for efficiently handling big data stored in heterogeneous devices of different networks are required. The MDC scheme proposed in this paper increases the throughput of cloud-based big data by predicting the data to be sent at later times and sending the difference of the consecutive data sequences, not the data itself, to the receiving node.

3.1. Overview. The MDC scheme transfers the difference between the consecutive data members, rather than the data member itself, in order to increase throughput and speed. Let $m[k]$ be the data member of a data set transmitted in a big data environment. Instead of sending $m[k]$ to the receiver, the MDC sends $d[k]$, the difference of the two consecutive data members, as represented in (1). k denotes the number of data members in the data set transmitted in a big data environment:

$$d[k] = m[k] - m[k-1]. \quad (1)$$

When $d[k]$ and the previous data member $m[k-1]$ are known, $m[k]$ can be derived. Thus, the receiver is able to create $m[k]$ with the received difference $d[k]$.

As presented in Figure 1, the MDC increases throughput by predicting $m[k]$, the k_{th} data member. Given that an estimate of $m[k]$ is $\hat{m}[k]$, the difference with a data estimation error, that is, $d[k] = m[k] - \hat{m}[k]$, will be transferred. The receiver predicts $\hat{m}[k]$ with the past data members and creates $m[k]$ by adding the received $d[k]$ to the predicted $\hat{m}[k]$.

Suppose that the all-order derivatives of the data member at a random time point t , $m(t)$, are known. Using a Taylor series, $m(t + T_s)$ is expressed in (2). T_s denotes the time at which server S generates the data member:

$$m(t + T_s) = m(t) + T_s \dot{m}(t) + \left(\frac{T_s^2}{2!} \right) \ddot{m}(t) + \left(\frac{T_s^3}{3!} \right) \dddot{m}(t) + \dots \quad (2)$$

$$\approx m(t) + T_s \dot{m}(t) \quad \text{for small } T_s. \quad (3)$$

In (2), if the data member and its derivative values at time t are known, the data member at a later time $t + T_s$ can be predicted. Even if only the first derivative is known, an approximate estimation of the future data member is possible, as expressed in (3). Let $m[k]$ be the k_{th} data of $M(t)$. If $m(kT_s) = m[k]$, then $m(kT_s \pm T_s) = m[k \pm 1]$. With $t = kT_s$

FIGURE 1: A multidata connection structure using the prediction coefficient.

in (3), $\hat{m}[kT_s] \sim [m(kT_s) - m(kT_s - T_s)]/T_s$ is produced. This leads to

$$m[k+1] \sim m[k] + T_s \left[\frac{(m[k] - m[k-1])}{T_s} \right] = 2m[k] - m[k-1]. \quad (4)$$

An approximate value $m(t) + T_s \dot{m}(t)$ in (3) is further refined by increasing the number of terms on the right side. To perform the differentials of higher order, more past data are needed. The accuracy of prediction increases as the number of previous data members increases. The prediction is carried out using

$$m[k] \approx a_1 m[k-1] + a_2 m[k-2] + \dots + a_N m[k-N]. \quad (5)$$

In (5), $a_1 m[k-1] + a_2 m[k-2] + \dots + a_N m[k-N]$ denotes $\hat{m}[k]$, that is, an estimate of $m[k]$. $\hat{m}[k]$ is expressed in

$$\hat{m}[k] = a_1 m[k-1] + a_2 m[k-2] + \dots + a_N m[k-N]. \quad (6)$$

As N increases, the accuracy of prediction increases. The data member sent by the server is $m[k]$ and the data member received by the receiver is $\hat{m}[k]$.

3.2. Linear Mean Square Error Estimation. When two estimation errors x and y of the data member of interest are related, one of them can be used to derive the other. That is, y can be estimated based on the knowledge of x . An estimate $\hat{y}$ of y is also likely to have an estimation error, making it not quite the same as the actual estimation error y . The MDC finds the best estimate of the data member estimation error using the minimum mean square error (MMSE) criterion, $\overline{\epsilon^2}$. This is expressed in

$$\overline{\epsilon^2} = \overline{(y - \hat{y})^2}. \quad (7)$$

In general, $\hat{y}$, the best estimate of the data member estimation error y , is a nonlinear function of x . Assuming that $\hat{x} = 0$, $\hat{y}$ can be restricted to a linear function of x , as represented in

$$\hat{y} = ax. \quad (8)$$

$\overline{\varepsilon^2}$ is computed using

$$\begin{aligned}\overline{\varepsilon^2} &= \overline{(y - \hat{y})^2} = \overline{(y - ax)^2} \\ &= \overline{y^2} + a^2 \overline{x^2} - 2a \overline{xy}.\end{aligned}\quad (9)$$

To minimize $\overline{\varepsilon^2}$ of (9), (10) is executed. Consider

$$\frac{\partial \overline{\varepsilon^2}}{\partial a} = 2a \overline{x^2} - 2 \overline{xy} = 0. \quad (10)$$

a is computed using

$$a = \frac{\overline{xy}}{\overline{x^2}} = \frac{R_{xy}}{R_{xx}}. \quad (11)$$

In the equation above, $R_{xy} = \overline{xy}$, $R_{xx} = \overline{x^2}$, and $R_{yy} = \overline{y^2}$. By applying a of (11) to (12), $y - R_{xy}x/R_{xx}$ is produced. Consider

$$\varepsilon = y - ax = y - \frac{R_{xy}x}{R_{xx}}. \quad (12)$$

As shown in (13), $\overline{x\varepsilon}$ is $\overline{x(y - (R_{xy}/R_{xx})x)} = \overline{xy} - (R_{xy}/R_{xx})\overline{x^2}$. Consider

$$\overline{x\varepsilon} = \overline{x\left(y - \frac{R_{xy}}{R_{xx}}x\right)} = \overline{xy} - \frac{R_{xy}}{R_{xx}}\overline{x^2}. \quad (13)$$

In (13), $\overline{xy} = R_{xy}$ and $\overline{x^2} = R_{xx}$. With this, $\overline{x\varepsilon}$ is rewritten as

$$\overline{x\varepsilon} = R_{xy} - R_{xy}. \quad (14)$$

Equation (14) is in line with the orthogonality principle. x is orthogonal ("not related") to ε when the mean square error (MSE) is the minimum.

By the orthogonality theorem, (15) computes the MMSE. Consider

$$\begin{aligned}\overline{\varepsilon^2} &= \overline{(y - ax)^2} = \overline{(y - ax)y} - a \cdot \overline{\varepsilon x} = \overline{(y - ax)y} \\ &= \overline{y^2} - a \cdot \overline{yx} = R_{yy} - a \cdot R_{xy}.\end{aligned}\quad (15)$$

3.3. Estimation of the Data Members. If an estimate of the data member of interest, x_0 , is related with other n estimates of the data member, $x_1, x_2, \dots, x_N$, then x_0 can be estimated via the linear summation of $x_1 + x_2, \dots, x_N$. This is represented in

$$\hat{x}_0 = a_1 x_1 + a_2 x_2 + \dots + a_n x_n = \sum a_i x_i. \quad (16)$$

As expressed in (17), $\overline{\varepsilon^2}$ is computed using $[x_0 - (a_1 x_1 + a_2 x_2 + \dots + a_n x_n)]^2$. Consider

$$\overline{\varepsilon^2} = \overline{[x_0 - (a_1 x_1 + a_2 x_2 + \dots + a_n x_n)]^2}. \quad (17)$$

To minimize $\overline{\varepsilon^2}$ of (17), (18) is performed. Consider

$$\frac{\partial \overline{\varepsilon^2}}{\partial a_1} = \frac{\partial \overline{\varepsilon^2}}{\partial a_2} = \dots = \frac{\partial \overline{\varepsilon^2}}{\partial a_n} = 0. \quad (18)$$

$\partial \overline{\varepsilon^2} / \partial a_i$ takes the derivative of $\overline{\varepsilon^2}$ with respect to a_i , as represented in

$$\frac{\partial \overline{\varepsilon^2}}{\partial a_i} = \frac{\partial}{\partial a_i} [x_0 - (a_1 x_1 + a_2 x_2 + \dots + a_n x_n)]^2 = 0. \quad (19)$$

$\partial \overline{\varepsilon^2} / \partial a_i$ is rewritten in (20) by rearranging the differential calculus and the average. Consider

$$\frac{\partial \overline{\varepsilon^2}}{\partial a_i} = -2 \overline{[x_0 - (a_1 x_1 + a_2 x_2 + \dots + a_n x_n)] x_i} = 0. \quad (20)$$

Equation (20) is rewritten in

$$\overline{\varepsilon \cdot x_i} = 0 \quad i = 1, 2, \dots, n. \quad (21)$$

Equation (21) can be transformed to (22) but the condition in (23) must be satisfied. Consider

$$R_{0i} = a_1 R_{i1} + a_2 R_{i2} + \dots + a_n R_{in}, \quad (22)$$

but

$$R_{ij} = \overline{x_i x_j}. \quad (23)$$

By differentiating $\overline{\varepsilon^2}$ with regard to $a_1, a_2, \dots, a_n$ and equating the result to zero, n equations are obtained. Constant $a_1, a_2, \dots, a_n$ can be obtained using the inverse matrix, as shown in

$$\begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_n \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & \dots & R_{1n} \\ R_{21} & R_{22} & \dots & R_{2n} \\ \dots & \dots & \dots & \dots \\ R_{n1} & R_{n2} & \dots & R_{nn} \end{bmatrix} \begin{bmatrix} R_{01} \\ R_{02} \\ \dots \\ R_{0n} \end{bmatrix}. \quad (24)$$

In (20) and (21), the error ε is orthogonal to the data member estimates $x_1, x_2, \dots, x_n$ so as to have the best estimate of the data member. The MSE in this optimum condition is expressed in

$$\overline{\varepsilon^2} = \overline{\varepsilon \varepsilon} = \overline{\varepsilon [x_0 - (a_1 x_1 + a_2 x_2 + \dots + a_n x_n)]}. \quad (25)$$

In (25), $\overline{\varepsilon x_i}$ is 0 ($i = 1, 2, \dots, n$). $\overline{\varepsilon^2}$ is rewritten in

$$\begin{aligned}\overline{\varepsilon^2} &= \overline{\varepsilon x_0} = \overline{x_0 [x_0 - (a_1 x_1 + a_2 x_2 + \dots + a_n x_n)]} \\ &= R_{00} - (a_1 R_{01} + a_2 R_{02} + \dots + a_n R_{0n}).\end{aligned}\quad (26)$$

4. Experiments

4.1. Performance Measures. In the experiments, MATLAB was used to evaluate the proposed MDC scheme with $\overline{\varepsilon^2}$, the MMSE of data member estimations. For simplification, it was

assumed that the data member $m[k]$ has the same attribute function. With this assumption, all data members have the same size and the receiver needs the same storage space for $m[k]$. If an estimate of the data member $m[k]$ is $\hat{m}[k]$, the difference with an estimation error, $d[k] = m[k] - \hat{m}[k]$, is transmitted.

The performance of the MDC was evaluated in terms of data member prediction accuracy measured using $P_{m[k]}$, the probability of mean squared data prediction error. To evaluate the accuracy of the received difference information, two random nodes n_i and n_j share the MSEs associated with the data differences. Here, the difference can be between two consecutive data members, denoted as $d[1] (= m[1] - m[0])$, and between as many as $q (= d[q])$. $P_{m[k]}$ is calculated using

$$P_{m[k]} = 1 - \left(1 - \frac{1}{k}\right)^m. \quad (27)$$

k denotes the predicted location of the data member used in the MSE. m denotes the number of data differences used in the MSE. m denotes the size of $\hat{m}[k]$:

$$\text{Prob}\{\bar{d} = j\} = \frac{d-1}{j-1} P_{m[k]}^{j-1} (1 - P_{m[k]})^{d-j}, \quad (28)$$

$$\text{Prob}\{\bar{d} = j\} = \frac{d}{j} P_{m[k]}^j (1 - P_{m[k]})^{d-j}. \quad (29)$$

In the equations above, $\bar{d}$ denotes the probability of MSE of $\hat{m}[k]$. $\bar{d}$ chooses m of $\hat{m}[k]$ that satisfies $1 \leq j \leq d$. Using j , $1 + (d-1) \cdot P_{m[k]}$ is produced. $\bar{d}$ of $\hat{m}[k]$ is j that obtains the actual data member $m[k]$ and satisfies $0 \leq j \leq d$.

To measure the performance of the proposed MDC scheme, $\bar{\varepsilon}^2$, $\hat{m}[k]$, and $\bar{d}$ were used.

4.2. Experiment Results. Figure 2 shows the data accuracy in terms of $\bar{\varepsilon}^2$ where $P_{m[k]}$ is 0.5 and k varies from 50,000, 100,000, 250,000, up to 500,000. Compared to the previous scheme that sends the data member itself, the MDC decreases the amount of power necessary for the data to be sent from the server to the receiver by 28%. As the number of data members increases, the throughput and storage space efficiency of the MDC increases.

Figure 3 shows $P_{m[k]}$ of the data members that are randomly selected from the networks of different sizes. The results show that $P_{m[k]}$ increases as the network size increases. These results were obtained by analyzing many different $\bar{\varepsilon}^2$, $\hat{m}[k]$, and $\bar{d}$ values. In Figure 3, $P_{m[k]}$ is proportional to $2/k$.

Figure 4 presents the delays in obtaining the data at the receiver based on the probability of prediction errors. Compared to the previous scheme in which the data is sent directly by the server to the receiver, the MDC decreases the delay by 28.6%. This indicates that transferring the data difference instead of the data itself contributes to providing seamless big data services to the end users.

Figure 5 shows the service delays with regard to the number of hops between source (the content server) and destination (the user). In comparison with the previous

FIGURE 2: Accuracy in terms of the linear MSE.

FIGURE 3: The probability of data prediction errors.

FIGURE 4: Delays in obtaining the data.

FIGURE 5: Service delays with regard to hop counts.

scheme, the proposed MDC has a constant service delay irrespective of the number of network hops. The MDC reduces the amount of data to be transferred by delivering only the difference of data sequences, and thus it is less affected by different network conditions.

5. Conclusion

This paper presented the MDC scheme that improves throughput by transferring the difference of the consecutive data sequences, instead of the data itself, in heterogeneous big data environments. The MDC minimizes the data transfer time between the content server and the user so that the user who moves across different networks can receive big data services without interruption. As it sends only the difference of the consecutive data members, high throughput is achieved irrespective of the types, functionalities, and characteristics of the data.

In comparison with the previous scheme, it was observed that the MDC decreases the amount of power required to communicate the data by 28%. This processing efficiency increases as the number of data members to be delivered increases, which leads to increased storage space efficiency. The MDC decreases the delay in obtaining the data at the receiving node by 28.6%. In addition, the MDC maintains a constant service delay irrespective of the number of network hops. In the future, the performance of the MDC will be examined in an environment where two or more networks are interacting with, and its resilience against various security attacks will be studied.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in *Proceedings of the Conference Record of the 36th Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 240–244, Pacific Grove, Calif, USA, November 2002.
- [2] A. Aaron, S. Rane, E. Setton, and B. Girod, "Transform-domain Wyner-Ziv codec for video," in *Visual Communications and Image Processing*, vol. 5308 of *Proceeding of SPIE*, p. 520, 2004.
- [3] A. Aaron, S. Rane, and B. Girod, "Wyner-Ziv video coding with hash-based motion compensation at the receiver," in *Proceedings of the International Conference on Image Processing (ICIP '04)*, vol. 5, pp. 3097–3100, October 2004.
- [4] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The discover codec: architecture, techniques and evaluation," in *Proceedings of the 26th Picture Coding Symposium (PCS '07)*, pp. 1–4, November 2007.
- [5] J. Ascenso and F. Pereira, "Adaptive hash-based side information exploitation for efficient Wyner-Ziv video coding," in *Proceedings of the 14th IEEE International Conference on Image Processing (ICIP '07)*, pp. III29–III32, IEEE, San Antonio, Tex, USA, September 2007.
- [6] M. Asif and J. J. Soraghan, "Wyner Ziv codec design for surveillance system using adaptive puncturing rate," in *Proceedings of the 3rd International Symposium on Communications, Control, and Signal Processing (ISCCSP '08)*, pp. 1454–1459, IEEE, March 2008.
- [7] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *Proceedings of the 13th Meeting of Video Coding Experts Group (VCEG '01)*, Austin, Tex, USA, April 2001.
- [8] G. Côté, B. Erol, M. Gallant, and F. Kossentini, "H.263+: video coding at low bit rates," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 7, pp. 849–866, 1998.
- [9] F. Dufaux, W. Gao, S. Tubaro, and A. Vetro, "Distributed video coding: trends and perspectives," *EURASIP Journal on Image and Video Processing*, vol. 2009, Article ID 508167, 2009.
- [10] C. Brites and F. Pereira, "Encoder rate control for transform domain Wyner-Ziv video coding," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '07)*, vol. 2, pp. II-5–II-8, San Antonio, Tex, USA, September-October 2007.