Multi-source Heterogeneous Data Acquisition Algorithm Design Different Time Periods

Jun Li(✉) and Jun Xing

Shenzhen Academy of Inspection and Quarantine, Shenzhen, China
zwm20171028@sina.com

Abstract. The traditional algorithm was affected by dynamic error and data loss, resulting in low efficiency of collection. In order to solve this problem, a time division collection algorithm based on data format transformation was proposed. According to the data format conversion process multi-source heterogeneous configuration files, and access to the content of the whole configuration file and the GDAL, according to the results of the configuration process design algorithm, under the constraints of the input data for approximate operation, minimize the objective function, through the fixed matrix other factors influence on partial derivatives root, period of time the multi-source heterogeneous data acquisition algorithm design. The experimental results showed that the maximum collection efficiency of the algorithm can reach 90%, which provided an effective solution for scientific researchers to solve the problems caused by differences in data format.

Keywords: Multiple source · Heterogeneous data · Period of time · Acquisition · Dynamic error · Packet loss

1 Introduction

Information physics fusion system has a wide range of applications, and the complexity of object orientation determines that the information it obtains in the physical world is heterogeneous. In recent years, information acquisition means have been increasing, such as ocean monitoring instruments, ocean satellite remote sensing, global buoy program, numerical model calculation, etc. On the one hand, the data obtained by these means enriches the data information and provides a very favorable foundation for research in various fields. On the other hand, it also brings about a variety of problems in data formats in various fields. Therefore, how to read these data is one of the primary problems that researchers need to solve [1].

In view of this problem, a multi-source heterogeneous data integration system is constructed by using traditional algorithm. However, users can only access the data in the system, but still cannot process their own data formats.

For the above problems and the deficiencies of existing integrated systems, a time-division acquisition algorithm of multi-source heterogeneous data is proposed [2]. For project developers, this can reduce the workload of development and shorten the development cycle, and for users or data researchers, it is convenient for them to conduct follow-up processing of data.

G. Gui and L. Yun (Eds.): ADHIP 2019, LNICST 301, pp. 104–111, 2019.
https://doi.org/10.1007/978-3-030-36402-1_11

2 Based on Data Format Conversion Algorithm Design

Since the storage mechanism of multi-source heterogeneous data format is different, some data formats can be read through the same third-party library function, so they can be processed together [3]. The entire data format transformation process is shown in Fig. 1.

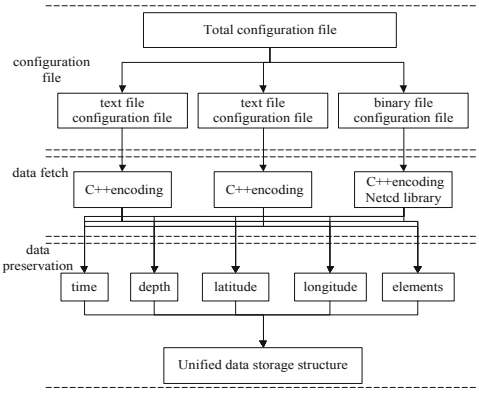

Fig. 1. Data format conversion process

It can be seen from Fig. 1 that when using the data conversion system to process data, the multi-source heterogeneous data format is divided into 4 categories for interpretation, which are text file classes, mainly including text format files. Binary file classes, mainly including binary format files; GDAL files, mainly including NetCDF format files, rattle format files, HDF format files (including HDF4 and HDF5 formats), remote sensing image files; MATLAB file, mainly including Mat format file [4].

2.1 File Configuration

The configuration files in the system are divided into two parts: one is the general configuration file, whose purpose is to convey what type of data the system will analyze next; The other part is the sub-configuration file corresponding to the 4 class files. The sub-configuration file mainly completes the description of each file class [5]. Take the GDAL file class as an example to detail the specific contents of the configuration file:

(1) the general configuration file is as follows:

File type: 2 (default in this system: the number “0” stands for text file class, the number “1” stands for binary file class, the number “2” stands for GDAL file class, and the number “3” stands for MATLAB file class);

(2) the sub(GDAL file class) configuration file is as follows:

File path: that is, the path of the file to be read (you can choose to read individual files or all files under a folder);

Depth file: that is, the depth file corresponding to the file to be read. If it exists, write the depth file name; otherwise, write "0";

① Name of the time variable: the name of the time set in the file to be read;

Time range: that is, the data reader can freely choose the time range of reading within the time range of the file to be read;

② Name of the depth variable: the name of the depth set in the file to be read;

Depth range: that is, the data reader can freely choose the depth range to read within the depth range of the file to be read;

③ Name of the latitude variable: the name of the middle latitude set of the file to be read;

Latitude range: that is, the data reader can freely choose the latitude range to read in the latitude range of the file to be read;

④ Name of the depth variable: that is, the name of the depth set in the file to be read;

Depth range: that is, the data reader can freely choose the depth range to read within the depth range of the file to be read;

⑤ Name of the slave latitude variable: that is, the name of the middle latitude set of the file to be read;

Latitude range: that is, the data reader can freely choose the latitude range to read in the latitude range of the file to be read;

⑥ Longitude variable name: the name of longitude set in the file to be read;

Longitude range: that is, the data reader can freely choose the longitude range read within the longitude range of the file to be read;

⑦ The name of a scheduled element: the name of the set of variables to be read in the file to be read.

⑧ Proportion factor: the scale factor of the set of variables to be read;

⑨ Null value: that is, invalid value data in the set of variables to be read;

⑩ Bonus units: units of the set of variables to be read [6].

2.2 Algorithm Design

In the case of data format transformation, the multi-source heterogeneous file is configured and the time-division acquisition algorithm of multi-source heterogeneous data is designed according to the configuration results [7]. The parameters and operation symbols used in the algorithm are as follows:

λ is the set of all relational matrices Rxy, where x,y $\in \{1, \ldots, r\}$ is; Z^t constraint matrix, where t $\in \{1, 2, \ldots, \max_x t_x\}$; The rank of p_q matrix, where $q \in \{1, \ldots, r\}$; A, B factor matrix.

The specific design of algorithm flow is as follows:

Start with x from 1 to r to initialize BX.

Repeat the following process until convergence:

Construct matrix C and D according to the above definition.

Obtain the value of E according to the calculation result of formula $E = (B^T B)^{-1} B^T C B (B^T B)^{-1}$.

Start x from 1 to r, and set B_x^e to the 0 matrix.

Start x from 1 to r, and set B_x^{d} to the 0 matrix.

For each relation matrix ruler C_{xy} belonging to λ, perform the following operations:

$$B_x^e += \left(C_{xy} B_y E_{xy}^T\right)^+ + B_x \left(E_{xy} B_y^T B_y E_{xy}^T\right)^- \tag{1}$$

$$B_x^{\mathrm{d}} += \left(C_{xy} B_y E_{xy}^T\right)^- + B_x \left(E_{xy} B_y^T B_y E_{xy}^T\right)^+ \tag{2}$$

$$B_x^e += \left(C_{xy}^T B_y E_{xy}^T\right)^+ + B_x \left(E_{xy}^T B_y^T B_y E_{xy}\right)^- \tag{3}$$

$$B_x^{\mathrm{d}} += \left(C_{xy}^T B_y E_{xy}^T\right)^- + B_x \left(E_{xy}^T B_y^T B_y E_{xy}\right)^+ \tag{4}$$

T to 1 to $max_x t_x$, do the following:

X from 1 to r, do the following:

$$B_x^e += \left[\theta_x^t\right]^- B_x \tag{5}$$

$$B_x^{\mathrm{d}} += \left[\theta_x^t\right]^+ B_x \tag{6}$$

constructing matrix structure:

$$B = B \circ Diag\left(\sqrt{\frac{B_1^e}{B_1^d}}, \sqrt{\frac{B_2^e}{B_2^d}}, \ldots, \sqrt{\frac{B_r^e}{B_r^d}}\right) \tag{7}$$

For the initialization of each B_x, the random hcol algorithm is adopted in this algorithm: the value of each column of B_x is calculated by averaging the elements of the random subset of the column in ruler C_{xy}. And for the matrix A, you don't have to initialize it, because it can be calculated from the value of the matrix B.

The time-division acquisition algorithm based on data format transformation performs approximate operation on the input data according to the constraint conditions to minimize the objective function, where the objective function is:

$$\min_{B \geq 0} J(B : E) = \sum C_{xy} \in \lambda \left\| C_{xy} - B_x E_{xy} B_y^T \right\|^2 \tag{8}$$

The functions II*ll and tr(*) respectively represent the f-normal form and the trace function, and λ is the set of all relationships between objects. The missing relational matrix is replaced by the 0 matrix. Although it can achieve the purpose of the objective function optimization, it will also bring an unexpected relational matrix in the factorization and distort the relationships between objects.

To solve the minimization problem, the factor matrix needs to be initialized first, and then keep E unchanged, change B or keep B unchanged, change the value of E and iterate over them until the expression converges. Changing the values of B_x and E_{xy} is the local optimum of their convergence to the optimization problem. By fixing the value of one of B and E and the influence of Lagrangian other matrix factors on partial derivative roots, the multiplication and updating rules of relational matrix were changed, thus completing the design of time-division acquisition algorithm for multi-source heterogeneous data.

3 Experiment

To test the validity of the algorithm based on the data format conversion, the following experiment has been conducted.

3.1 Experiment Platform Design

NS2 and MATLAB r2000b are used as experimental platforms. NS2 is a powerful network simulation platform, while MATLAB has a powerful matrix processing function. NS2 is an object-oriented network simulator, which is essentially a discrete event simulator. NS2 itself has a virtual clock, and all events are driven by discrete events. At present, NS2 can be used for simulation of different IP networks, and some simulations have been implemented: (1) network transmission protocols such as TCP/UDP; (2) generate business source traffic; (3) routing queue management mechanism, etc. Since NS2 is open source, some research groups continue to enrich the component library, making it more advantageous. As shown in Fig. 2, NS2 structure diagram introduces each component.

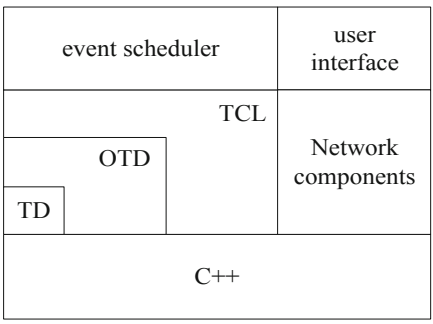

Fig. 2. NS2 structure diagram

The development language of NS2 is C++ and Otcl. Objects and variables in both C++ and Otcl are related by Tcl. C++ classes and objects are compiled classes and objects, whereas OTcl classes and objects are called explanatory classes and explanatory objects.

3.2 Experimental Results and Analysis

The traditional algorithm is compared with the time-division acquisition algorithm based on data format conversion in the case of dynamic error and packet loss, and the results are as follows.

3.2.1 Dynamic Error

The comparison and analysis results of the two algorithms under dynamic error are shown in Fig. 3.

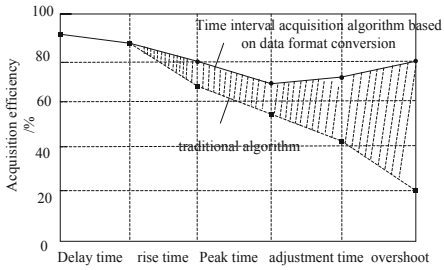

Fig. 3. Comparison and analysis results of acquisition efficiency of the two algorithms under dynamic error

As can be seen from the trend of broken lines in Fig. 3, the accuracy of the initial analysis results of the two algorithms can reach 90%. When the dynamic response is within the delay period, the efficiency of the traditional algorithm is the same as that of the time-division acquisition algorithm based on data format transformation, both of which are 85%. When the dynamic response is within the rising period, the collection efficiency of the traditional algorithm is affected by the regional environment, resulting in a low collection efficiency of 65%. Although the time-division acquisition algorithm based on data format transformation will not be affected by the regional environment, the collection efficiency will be reduced to 80% as the experiment is controlled under the dynamic environment. When the dynamic response is within the peak period, the collection efficiency of the traditional algorithm reaches 55%, while the collection efficiency of the time-division acquisition algorithm based on data format transformation can reach 70%. When the dynamic response is within the adjustment period, the adjustment effect of the traditional algorithm is poor, resulting in the acquisition efficiency falling to 41%. The efficiency of time - segment acquisition algorithm based on data format conversion increased to 75%. When the dynamic response is in the overshoot process, the acquisition efficiency of the traditional algorithm has reached the

minimum, at 20%. And the efficiency of the time - segment acquisition algorithm based on data format conversion continues to increase, reaching 80%. According to the analysis results, the traditional algorithm is affected by the regional environment and cannot effectively analyze errors in the dynamic response process, resulting in a low collection efficiency. The time - segment acquisition algorithm can avoid the influence of regional environment and maintain high efficiency.

3.2.2 Packet Loss

The comparison and analysis results of the acquisition efficiency of the two algorithms in the case of packet loss are shown in Fig. 4.

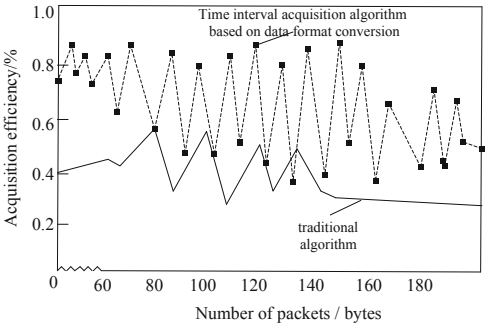

Fig. 4. Comparison and analysis results of acquisition efficiency of the two algorithms in the case of packet loss

According to the comparison diagram in Fig. 4, when the number of lost packets is less than or equal to 60 bytes, the maximum collection efficiency of the time-division acquisition algorithm based on data format conversion is 0.87, while the maximum value of the traditional algorithm is 0.43. When the number of packets is greater than 60 and less than or equal to 143 bytes, the maximum collection efficiency of the time-division acquisition algorithm based on data format conversion is 0.9, while the maximum value of the traditional algorithm is 0.6. When the number of lost packets is greater than 143 and less than 160 bytes, the maximum collection efficiency of the time-division acquisition algorithm based on data format conversion is 0.89, while the maximum value of the traditional algorithm is 0.4. When the number of lost packets is greater than or equal to 143 and less than 200 bytes, the maximum collection efficiency of the time-division acquisition algorithm based on data format conversion is 0.82, while the maximum value of the traditional algorithm is 0.39. According to the analysis results, when packet loss occurs in the network, the time-division acquisition algorithm based on data format conversion is more efficient.

3.3 Experimental Conclusions

According to the packet loss situation, the two algorithms can obtain the following comparison results: when the number of packet loss is less than or equal to 60 bytes,

the maximum collection efficiency difference between the two algorithms is 0.44. When the number of lost packets is greater than 60 and less than or equal to 143 bytes, the maximum collection efficiency of the two algorithms differs by 0.3. At that time, when the number of packet loss was greater than 143 and less than 160 bytes, the maximum value difference of acquisition efficiency of the two algorithms was 0.49. When the number of lost packets is greater than or equal to 143 and less than 200 bytes, the maximum collection efficiency of the two algorithms differs by 0.43. Therefore, the design of time - segment acquisition algorithm based on data format transformation is reasonable.

4 Conclusions

The design method of time-division acquisition algorithm based on data format transformation can still maintain a good collection efficiency under the problems of dynamic error and data loss. By summarizing the typical cases in the daily combined test, it can be seen that the maximum difference between the collection efficiency of the algorithm and that of the traditional algorithm is 0.71, which has significant application effect.

Acknowlegements. National Key R&D Program of China (2017YFF0211100).
Shenzhen Science and Technology Project (KJYY20160229141621130).

References

1. Liu, S., Li, N., Fu, J.: A peak-valley time division model based on high-dimensional norming and SGHSA algorithm. China Electr. Power **51**(1), 179–184 (2018)
2. Li, B., Huang, J., Wu, Y., et al.: Short-term load forecasting of typhoon based on meteorological information particle reduction. J. Electr. Technol. **33**(9), 2068–2076 (2018)
3. Wang, S., Shi, C., Qian, G., et al.: Chaotic time series prediction based on fractional order maximum correlation entropy algorithm. J. Phys. **20**(1), 248–255 (2018)
4. Liu, C., Hu, N., Guo, Z., et al.: Numerical simulation of wave field in viscous fluid biphasic VTI medium based on fractional time derivative constant Q viscoelastic constitutive relation. Geophysics **19**(6), 24–25 (2018)
5. Shi, J., Zhang, J.: Vehicle routing problem model and algorithm for batch distribution with stochastic travel time. Comput. Appl. **38**(2), 573–581 (2018)
6. Wang, Z., Yi, Lin, Lin, Y.: Similarity measurement of time series based on coefficient matrix arc differential. Comput. Eng. **20**(2), 9–16 (2018)
7. Wang, Y., Lian, C.H., Jin, Q.: Single quantum bit storage time refreshes the world record - single ion qubit. Physics with more than 10 minutes of coherent time, **47**(5), 320–322 (2018)