Studies of an Event Building algorithm of the readout system for the twin TPCs in HFR

Authors: Hong-Yun Zhao, Zheng-Guo Hu, Xi-Meng Chen, Jing Tian, Zhi-Peng Sun, Song-Bo Chang, Yi Qian, Yi Qian

Date: 2024-04-04T12:11:54+00:00

Abstract

The High Energy Fragment Separator (HFRS), which is currently under construction, is a leading international radioactive beam device. Multiple sets of position-sensitive Twin Time Projection Chamber (TPC)detectors are distributed on HFRS for particle identification and beam monitoring. The twin TPCs' readoutelectronics system operates in a trigger-less mode due to its high counting rate, leading to a challenge of handling large amounts of data. To address this problem, we introduced an event-building algorithm. This algorithmemploys a hierarchical processing strategy to compress data during transmission and aggregation. In addition,it reconstructs twin TPCs' events online and stores only the reconstructed particle information, which significantly reduces the burden on data transmission and storage resources. Simulation studies demonstrated that thealgorithm accurately matches twin TPCs' events and reduces more than 98% of the data volume at a countingrate of 500 kHz/channel.

Full Text

Abstract

Studies of an Event Building algorithm for the readout system of twin TPCs in HFRS

Jing Tian,[1,2,3,4] Zhi-Peng Sun,[1,3] Song-Bo Chang,[1,3,4] Yi Qian,[1,3,4,†] Hong-Yun Zhao,[1,3,4] Zheng-Guo Hu,[1,3,4] and Xi-Meng Chen[2]

[1]Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou 730000, China

[2]School of Nuclear Science and Technology, University of Lanzhou, Lanzhou 730000, China

[3]School of Nuclear Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China

[4]Advanced Energy Science and Technology, Guangdong Laboratory, Huizhou 516000, China

The High Energy Fragment Separator (HFRS), currently under construction, represents a state-of-the-art international radioactive beam facility. Multiple sets of position-sensitive Twin Time Projection Chamber (TPC) detectors are deployed throughout HFRS for particle identification and beam monitoring. Due to the high counting rates, the twin TPC readout electronics system operates in trigger-less mode, creating significant challenges in handling massive data volumes. To address this issue, we have introduced an event-building algorithm that employs a hierarchical processing strategy to compress data during transmission and aggregation. Furthermore, the algorithm reconstructs twin TPC events online and stores only the reconstructed particle information, substantially reducing the burden on data transmission and storage resources. Simulation studies demonstrate that the algorithm accurately matches twin TPC events and reduces data volume by over 98% at a counting rate of 500 kHz/channel.

Keywords: High counting rate, Twin TPCs, Trigger-less, Readout electronics, Event building, Hierarchical data processing

INTRODUCTION

With the continuous advancement of superconducting heavy-ion accelerator technology, developed countries are proposing the construction of next-generation (third-generation) large scientific facilities to obtain higher-intensity and higher-energy radioactive nuclear beams, aiming to expand research into heavier nuclides closer to the drip line [**?**, **?**, **?**]. Currently, facilities such as BigRIPS [**?**] at RIKEN in Japan and ARIS [**?**] at MSU in the United States have been completed and are operational, while others including SuperFRS [**?**] at GSI in Germany and the High Energy Fragment Separator (HFRS) [**?**, **?**, **?**, **?**] at the Institute of Modern Physics, Chinese Academy of Sciences, are under construction. Upon completion, HFRS will represent a new generation of radioactive beam facilities with higher beam energy and magnetic rigidity (B), significantly enhancing experimental capabilities in the medium-heavy nuclear region.

Given the high-energy and high-intensity characteristics of HFRS, highly reliable nuclide identification at high counting rates (10 MHz) and large dynamic range requirements (Z = 1   92) is necessary, posing substantial challenges for radioactive particle detection and readout electronics technology. The Time Projection Chamber (TPC) [**?**, **?**, **?**, **?**] is a highly efficient, high-resolution particle tracking detector that enables reconstruction of particle trajectory, momentum, and energy loss. It plays a crucial role in particle physics research and has been widely employed in nuclear physics experiments such as the ALICE [**?**, **?**, **?**] heavy-ion experiment at CERN, the RHIC-STAR [**?**, **?**] experiment at Brookhaven National Laboratory, and the CSR External-Target Experiment (CEE) [**?**, **?**]

currently under construction in China. Due to its high reliability, the TPC has been selected as a position-sensitive detector for particle identification and beam monitoring.

However, as projectile counting rates increase, there is a strong possibility that readout signals from different particles become temporally confused. For instance, at a counting rate of 10 MHz, with a drift distance of 5 cm and drift velocity of 5 cm/ s, particles arrive at the detectors every 100 ns while the maximum drift time is 1 s. Under these conditions, the probability that signals from a later-arriving particle are sensed on the readout strips before those from an earlier particle reaches 90%. To address this issue, two TPC detectors with identical configurations are placed in close proximity and inverted to form a set of twin TPCs [?], as illustrated in Fig. 1. When the same beam particle passes through the twin TPCs, electrons generated by ionization produce induced signals at the anode readout strips that carry both energy and time information. Once the readout electronics acquire and process these induction signals, the hit position of the incident particle and the electron drift times in both TPCs become available for further analysis.

In low counting rate experiments such as GET (1 kHz) [?] and medium counting rate experiments such as CEE (10 kHz) [?, ?], readout electronics employ trigger-based methods where server clusters complete data processing tasks including transmission, event building, and storage. However, as counting rates increase, the trigger-based approach may no longer satisfy experimental requirements. Consequently, a trigger-less readout scheme [?, ?] has been proposed. For example, the ATLAS (40 MHz total collision event rate) [?] and ALICE (50 kHz) [?] upgrade projects have adopted trigger-less readout systems, enabling analysis and filtering of massive data volumes in the back-end. Compared with conventional trigger-based systems, trigger-less systems collect all raw data from front-end electronics, improving the efficiency of obtaining valid events during experiments while implementing trigger algorithms in software, thus providing greater flexibility. However, trigger-less systems require significantly higher bandwidth for transmission and processing, as well as increased computing power, which raises development difficulty and overhead for readout systems.

To overcome the bottleneck of data transmission and storage in trigger-less mode and accurately match events from the twin TPCs, we propose an online event-building algorithm. Our algorithm combines software and hardware to handle massive data volumes using a hierarchical processing strategy, addressing the issue of particle temporal disorder at high counting rates. Specifically, data are compressed in each electronics readout unit by extracting only key information, significantly reducing the burden on transmission and storage. Additionally, by utilizing particle hit position and drift time information obtained from the twin TPCs, we can effectively screen and reconstruct events, thereby recovering the original position information of incoming particles. Consequently, our readout system achieves both high counting rate capability and low data volume.

II. THE HIERARCHICAL DATA PROCESSING STRATEGY

The readout requirements for twin TPCs are listed in Table 1. Even if front-end electronics perform zero compression [?, ?], high counting rates place enormous pressure on the data acquisition system. Therefore, it is imperative to explore a hierarchical processing strategy within each readout unit to reduce transmission and storage pressures.

The twin TPC readout system employs a three-tiered structural design consisting of front-end electronics (FEEs), slave data acquisition units (slave DAQs), and a server, which facilitate data processing at the channel, detector, and subsystem levels, respectively. The schematic in Fig. 2 illustrates the hierarchy of the data processing system. The master data acquisition unit (master DAQ) of HFRS collects data from each subsystem server for downstream processing and storage. Because our work focuses on the subsystem level, we do not discuss the master DAQ in detail in this paper.

The primary function of the FEE is to process detector signals at the channel level. FEE components include front-end amplifiers for MWDC (FEAMs) [?, ?], which have been used in CEE, multichannel analog-to-digital converters (ADCs) (ADS52J90) [?], and a field-programmable gate array (FPGA) (Xilinx Kintex-7-325T) [?]. A FEAM chip can handle eight channels over a wide dynamic range (100 times). Each channel comprises a charge-sensitive amplifier (CSA), a shaper circuit, and a non-inverting driver circuit. The shaper circuit performs pole-zero cancellation, filtering, and shaping. The FEAM chip first amplifies and shapes weak signals from the readout strips, then passes the output to the ADC. The chip counting rate is currently being optimized for HFRS-TPC requirements. The design employs sub-board and mainboard configurations to facilitate readout capabilities for 32 or 64 channels. The sub-board comprises four FEAM chips, while the mainboard can house two sub-boards simultaneously. The FPGA firmware features an online algorithm for extracting time and energy information to minimize data volume.

The FPGA firmware of the FEE contains self-triggering, data packaging, baseline subtraction, zero-compression, and feature extraction modules that cooperate to implement initial data compression. The self-triggering module creates trigger signals for the data packing module at predetermined intervals, allowing the packing module to segment an uninterrupted input data stream based on these signals. Upon arrival of each trigger signal, a time window is initiated with duration equal to the time interval between triggers. Relevant data within the time window are packaged in a particular format. To minimize overhead caused by packet headers, a time window of 1000 ADC sampling intervals was utilized in this study. The system clock cycle counter is included in the packet header as a coarse timestamp supplied uniformly by the clock unit and calibrated at regular intervals. Subsequently, raw data are transmitted in packets. The baseline subtraction module adjusts the baseline by dynamically calculating baseline noise using currently available data. The zero-compression module discards invalid data by judging pulse waveform validity, then sends valid pulses to the feature extraction module to obtain high-precision pulse time and energy information. Finally, information for each pulse is repackaged and sent to the slave DAQ.

The slave DAQ executes detector-level data aggregation using a high-performance FPGA (Xilinx Kintex-UltraSCALE 060) [?], which aggregates and packages data from multiple FEEs via 10 Gbps high-speed optical fiber links. In trigger-less mode, the slave DAQ must sort data in chronological and channel order to facilitate downstream event screening and building. This is accomplished by developing polling and sorting modules within the FPGA firmware of the slave DAQ. The polling module systematically extracts data from eight or 16 FEEs, unpacks the data packets, and recovers original pulse information. This information is then transmitted to the sorting module, which arranges and orders all channel pulses using time information. Finally, the slave DAQ packages the sequenced data and transmits them to the server through a PCIe interface.

The server plays a crucial role in issuing slow control instructions for the electronics system and reconstructing subsystem events. The foremost step in event building involves using a time window to merge multiplicity signals. Once completed, the center of gravity (CG) approach determines the original incident positions of particles on the readout strips in each detector. Finally, both the incident position and drift time from the two TPCs are fed into the hit-matching module to reconstruct particle tracks. The amount of saved data is significantly minimized by removing invalid data.

III. DESIGN AND IMPLEMENTATION OF EVENT BUILDING ALGORITHM

There are two interpretations of event building [?]. Briefly, event building involves executing various software procedures on data, while in a broader sense, event building encompasses the system architecture of software-hardware collaboration. Specifically, multilevel processing—including data packaging, feature extraction, and aggregation—is first performed on the FPGAs of each readout electronics unit, after which software is leveraged to reconstruct physical events. Based on the hierarchical processing strategy outlined in Section II, the complete event-building algorithm flow includes signal preprocessing (adaptive baseline subtraction and zero compression), feature extraction (timing and energy extraction), multiplicity screening (fine timestamp sorting, fine timestamp merging, and CG for hit position), and hit matching. Figure 3 illustrates the detailed algorithm pipeline, and Fig. 4 displays the data format at each stage. A detailed description of each module's implementation is provided below.

A. Signal Pre-processing

To accommodate discrepancies across channels, channel-wise baseline and noise must be calculated within each packet. The baseline is determined by averaging the first 10 sample points of each channel, and noise is calculated using the root-mean-square value of these points. During selection, if any sample point exceeds the minimum signal level required for electronics processing, the operation stops and the region is deemed an effective signal segment. The baseline and noise are recalculated by moving back 20 points (adjusted for pulse width). Considering both baseline and noise, the threshold is measured as the sum of the baseline and four times the noise. If a sample point exceeds the threshold, the threshold is subtracted; otherwise, sample points less than or equal to the threshold are replaced with zero.

Only signals with more than three consecutive sample points above the threshold are considered valid pulse signals. The packet payload records the total quantity of these over-threshold sample points (Clust L), the serial number of the first over-threshold point in the packet (Clust T), and the corresponding data (D). Multiplying Clust T by the ADC sampling interval (50 ns) gives the pulse offset time relative to the coarse timestamp in the packet header. Following zero compression, the total quantity of data (ChLen) in each channel includes all pulse data points and corresponding descriptive information (Clust L and Clust T). This approach significantly reduces data volume.

B. Feature Extraction

In trigger-based mode, event screening relies on trigger signals, whereas in trigger-less mode it depends on high-precision time information. Therefore, implementing a timing module is essential for enhancing the accuracy of the inferred Clust T. The constant fraction discriminator (CFD) [?] divides the original signal into two paths: one delays the signal for a certain time while the other inverts and attenuates it. By merging these two paths, constant fraction timing is converted to zero-crossing timing, effectively mitigating time walk effects. Because the ADC operates via discrete sampling, the over-threshold time derived from CFD is often imprecise. Therefore, two data points located before and after the over-threshold time (Va and Vb) are identified, and a linear equation or interpolation technique is applied to achieve more precise time measurement.

During initial software simulation validation, we derived a linear equation for the amplitude and time of superimposed signals based on Va and Vb: the exact moment of over-threshold corresponds to when the amplitude is zero. In the firmware, we use an interpolation method to approximate the threshold, following the principle of dichotomy [?]. We first calculate the mid-amplitude point (Vmid) based on Va and Vb by dividing the original time interval into two equal sub-regions (a and b). The original time interval is equivalent to the ADC sampling interval. If Vmid multiplied by Vb is negative, we determine that the over-threshold moment lies in sub-region b; otherwise, it lies in the other sub-region. Thus, we reduce the time interval corresponding to the over-threshold moment to half the original. By repeating this process n times, we can increase timing accuracy by a factor of 2^n compared to the original. Finally, the timing result is superimposed on Clust T to provide more precise pulse time. Clust T is a 16-bit variable; its upper 10 bits record the serial number of data points within the packet (ranging from 0 to 1000) as the offset time relative to the coarse timestamp, while the remaining six bits record high-precision timing results, subdividing 50 ns into 64 equal parts to improve time resolution.

At high counting rates, events can accumulate. The peak method better distinguishes overlapping signals compared to area integration. When several adjacent sample points first increase then decrease, the peak (Clust E) is at the vertex. After energy extraction, only Clust T and Clust E are recorded for each pulse. We use the ChLen field to denote the pulse count for each channel to facilitate subsequent sorting.

C. Multiplicity Screening

When particles enter the detector, there is typically an angle between the incoming direction and detector plane. Transverse diffusion occurs after particle incidence, enabling multiple readout strips to produce induction signals. These signals have consistent timestamps but different magnitudes across multiple readout channels, a phenomenon called particle multiplicity [**?**] as shown in Fig. 5(a). Thus, multiplicity screening is required to recover the original hit position of the particle.

When merging particle multiplicity, the first step is to arrange pulses from all channels within the packet in chronological and channel order. Because all pulses in a packet share the same coarse timestamp (TCode), they only need to be sorted according to Clust T, which is known as fine timestamp sorting. Figure 5(b) shows the workflow of fine timestamp sorting. Since pulse data for each channel are stored in separate containers (FIFOs or arrays) in chronological order, only the first pulse data in each container are considered for comparison. Subsequently, the pulse data corresponding to the smallest moment obtained from this comparison are moved from the old container to the new container. In the second round, the second pulse data from the old container are compared with the first pulse data from other containers. This process continues until all data in each container are compared and stored in the new container. If pulse times in two containers are equal, they are arranged in channel order. If the time interval between adjacent channel pulse signals is shorter than the designated time window, they are categorized as the same event and can be merged according to channel order, as illustrated in Fig. 5(c). The Clust T value is determined based on the fine timestamp of the central channel, and a count field is added to record the total number of pulses in each event. This information is then used in the CG method to identify the hit position of the incident particle.

Using CG (in Eq. (1)), the hit position (Xhit) of each particle on the readout strip is determined based on the previously obtained channel-fired ID and energy value. The channel-fired ID is used as the xi coordinate, and the energy value serves as the weight Ei. The electron drift time is measured as the sum of the coarse timestamp and fine timestamp. Both the hit position and drift time from the twin TPCs are relayed to the hit-matching module, and the two-dimensional trajectory of incoming particles is ultimately reconstructed. This process also helps discard invalid or irrelevant data.

$$X_{hit} = \frac{\sum x_i \cdot E_i}{\sum E_i}$$

D. Hit Matching

In Eq. (2), the drift distance (L) and drift speed (vdrift) of electrons for twin TPCs are predetermined. Consequently, the sum of the drift times (tcs) remains fixed and can be used as a constraint to achieve hit-matching [**?**] of twin TPCs. Although the plastic scintillator detector (T0) can provide a highly accurate reference time (t0), large fluctuations in particle flight time can negatively impact its accuracy. Moreover, t0 information is transferred directly to the master DAQ, and system-wise event building is achieved by combining data from subsystem servers, which may reduce timeliness. Considering these factors, we propose a novel hit-matching algorithm for reconstructing particle tracks without t0 information. This enables event building to be performed on the subsystem server, thus enabling additional data compression and easing the computational burden on the master DAQ.

$$t_{cs} = t_u + t_d - 2t_0 = \frac{L}{v_{drift}}$$

where tu, td are the absolute drift times of electrons from the twin TPCs and t0 is the reference time from detector T0.

The matching algorithm can be classified into two methods based on whether the factor t0 is incorporated: the relative time method and the absolute time method.

1) The relative time method (a) Search for all hits from twin TPCs in the range [t0 - 5σt, t0 + tdmax + 5σt] according to t0. (b) Time matching: For the searched hits, use their drift times and t0 to calculate tcs, then select hit combinations within 5σtcs. (c) Position matching: For hits that meet time matching conditions, choose the combination with the smallest hit distance. (d) Positioning: According to the drift time of the chosen hit, combined with drift distance and drift speed, calculate the hit position in the Y direction.

2) The absolute time method (a) Search for all hits from the second TPC in the range [tu - tdmax - 6σt, tu + tdmax + 6σt] according to tu. (b) Time matching: For the searched hits, calculate |tu - td| and select hit combinations within 12σt. (c) Position matching: For hits that meet time matching conditions, choose the combination with the smallest hit distance and minimize the energy difference. (d) Positioning: According to the drift time of the chosen hit, combined with drift distance and drift speed, calculate the hit position in the Y direction.

$$\sigma_{tcs} = \sqrt{2\sigma_t^2 + 4\sigma_{t0}^2}$$

where σt, σt0, σtcs are the time resolutions of the TPC detector, T0 detector, and tcs variable, respectively.

IV. VERIFICATION AND ANALYSIS

As development of the TPC and front-end electronics is still underway, we were unable to test the algorithm on the actual devices. Instead, we first conducted software simulations to verify the performance of the proposed event-building algorithm, followed by preliminary testing with existing laboratory equipment.

A. Simulation Study

We used the Monte Carlo method [**?**] to generate simulation packets for the twin TPCs. When particles enter the twin TPCs, electrons generated by ionization drift upward and downward, with drift time based on their hit position in the Y direction while the total drift distance remains unchanged. The fired readout channels are determined by the hit position on the readout strips, incidence angle, and transverse diffusion, with a maximum of five channels. Therefore, drift time and fired channels can be set according to the length and width of the TPCs' incidence cross-section. The FEAM chip exhibited a peak time of 160 ns and a falling edge at approximately 320 ns. Consequently, the CR-$(RC)^3$ characteristic equation is employed to derive accurate pulse data, and the signal amplitude range is defined by the selected ADC bit width. When the system counting rate reaches 1 MHz, a particle is expected to hit the twin TPCs every 1000 ns on average. Therefore, two sets of simulation waveform data were simultaneously produced every 1000 ns (t0). The amplitude, fired channels, and drift time (with t0 as reference) for each waveform were randomly generated to accurately mimic real scenarios. Baseline noise obtained from existing FEE measurements was added to the simulation data to enhance authenticity. In addition, simulation data were produced for each TPC using only one FEE consisting of 32 channels.

Figure 6 shows the overall distribution of simulation data for a pair of complete packets for the twin TPCs. Multiple simulation waveforms were created within a 50 s period, traversing 32 different channels with varying peak amplitudes and covering most of the ADC range (bit width of 10). Figure 7(a) shows the simulated waveforms of the twin TPCs' central channel fired at a specific time, which exhibit high similarity with only minor amplitude variations. The phase relationship between the two drift times covers three cases: overrun, approach, and lag, satisfying simulation requirements. Furthermore, Fig. 7(b) shows the simulated waveforms after baseline removal, which drop to zero. The two TPC packets contained 219 and 210 pulses, respectively, meaning an average of 6-7 pulses were generated on each readout channel in 50 s for both TPCs. This yields an average counting rate greater than 100 kHz/channel as expected, indicating that the simulation data closely resembles actual conditions.

After merging multiplicities, the first TPC collected 50 sets of event data and the second TPC recorded 49 sets. Subsequently, the hit positions and drift times of all event data were transmitted to the hit-matching module, which employed the absolute time method for matching. Forty-nine datasets were successfully matched, with only one event dataset discarded due to matching failure. Finally, based on the matched drift time pairs, drift distances were calculated and particle hit positions in the Y-direction were recovered, as depicted in Fig. 8. When an incident particle strikes the twin TPCs, three drift distance scenarios are possible: the first TPC's drift distance may be greater than, less than, or equal to that of the second TPC, but the sum of drift distances in both TPCs always equals the maximum drift distance. Figure 8 shows three typical cases: (a) when the incident particle hits the center of the twin TPCs and the two drift distances are equal; (b) when the particle impacts a position close to the first TPC's readout strip, the second TPC experiences maximum drift distance; and (c) if the hit is offset toward the second TPC's readout strip, the first TPC undergoes a larger drift distance than the second. Theoretically, the particle should hit the same position on the readout strips of both TPCs. However, in practice, slight deviation between the two hit positions is possible due to oblique incidence. Using the relative-time approach, the same datasets produced consistent results, as shown in Fig. 8.

To verify algorithm performance at high counting rates, the hit probability of the readout channel was adjusted from uniform to Gaussian distribution, increasing the probability of readout channels being fired in the central region. As illustrated in Fig. 9(a), four to five channels within the two TPC packets were fired more than 25 times in 50 s, indicating a counting rate of 500 kHz/channel. Three sets of packets were analyzed, and matching results obtained using both methods were consistent, as shown in Fig. 9(b). Both matching algorithms could effectively recover two-dimensional particle traces under normal conditions. However, the relative time method is more suitable for severe cases because it employs a more rigorous equation as a constraint. Therefore, the absolute time method can be utilized in the subsystem server to initially match hits, remove inconsistent data, and further reduce data volume, while the relative-time technique can be employed in the master DAQ to accomplish system-wise hit identification.

As an illustration, the current simulation parameters—a single data packet containing 32 channels with 1000 sample points each plus 16-bit descriptive information as raw data volume—were used to evaluate data compression capability. The evaluation was based on the following assumptions: (1) A total of 50 particles are incident in 50 microseconds, each hitting 5 readout channels, generating 250 valid pulses; (2) After baseline removal, an average of 15 sampling points are retained per valid pulse; (3) Fine timestamps and energy information are each 16 bits; (4) Hit position information is 10 bits in both X (readout strip) and Y directions.

The compression ratios for several stages are listed in Table 2, indicating that the event-building algorithm can shrink 99% of raw data and 97% of zero-compressed data.

B. Test Verification

As shown in Fig. 10, a primary test system was set up in the laboratory using existing electronic devices. Note that the chassis of the remote server and slave DAQ contained within it are not visible in this figure. A signal generator (Keysight 33522B) [**?**] was employed to create a trigger signal and two pulse signals. The two pulse signals were fed into the analog inputs of two FEAM chips on the FEE to emulate signals from two TPCs, respectively. Input signals were first converted to digital form by the ADC, then transmitted via optical fiber to the slave DAQ for aggregation and packaging. Finally, data were sent to a remote server through a PCIe interface. The drift time of electrons generated by primary ionizing particles was simulated by adjusting the delay time of the input signal relative to the trigger signal. Multiple pulse signals collected by the FEE were spliced together to mimic a trigger-less mechanism. With a sampling frequency of 50 MHz, the ADC collects 100 samples per pulse signal, implying a pulse period of 2 s and a counting rate of 500 kHz/channel.

Figure 11(a) illustrates the initial waveforms from two FEE channels' output at specific times. Next, the adaptive baseline subtraction module analyzes and removes baselines from the data, with results shown in Fig. 11(b), demonstrating the module's effectiveness in cleaning baseline noise.

To verify algorithm performance using experimental data, we selected eight typical delay time pairs and five different amplitude combinations of input signals to conduct 40 tests, each consisting of eight pulses. The additional delay caused by the readout system must be subtracted before data processing. After processing, 320 test pulses were sent to the hit-matching module, all of which were accurately matched using either absolute or relative time methods. Y-direction particle hit positions were determined using 320 pairs of drift times. From this dataset, 20% of the data were extracted and plotted in Fig. 12, showing that results from both matching algorithms were highly consistent. Key test parameters and compression ratios are shown in Table 3. Preliminary evidence demonstrates that the event-building algorithm can effectively compress data.

V. CONCLUSION

This paper presents an event-building algorithm designed to tackle two major challenges: the transmission and storage of large data volumes generated by trigger-less readout systems, and the issue of particle temporal disorder at high counting rates. The algorithm performs hierarchical processing and compression of data on FEEs, slave DAQs, and a server platform, finally matching twin TPC events on the server to reconstruct two-dimensional particle trajectories. The complete algorithm flow was implemented in software. Data from both simulations and preliminary laboratory tests were accurately matched using the algorithm, with results suggesting it can effectively reduce data volume by at least 98%. Potential future improvements include implementing firmware logic for timing, energy extraction, and sorting using FPGAs on both FEEs and slave DAQs, leveraging FPGA pipeline parallelism and low latency to enhance overall system performance. Additionally, employing GPU multicore parallel computing [?, ?] to improve the speed of multiplicity merging and hit matching modules represents another promising research direction.

AUTHOR CONTRIBUTIONS

All authors contributed to study conception and design. Algorithm implementation and material preparation were performed by Jing Tian and Zhi-Peng Sun. Data collection and analysis were performed by Jing Tian and Song-Bo Chang. The first draft of the manuscript was written by Jing Tian, and all authors commented on previous versions. All authors read and approved the final manuscript.

CONFLICT OF INTEREST

The authors declare that they have no competing interests.

REFERENCES

[1] Y. Zhang, D.W. Zhang, X.F. Luo, Experimental study of the QCD phase diagram in relativistic heavy-ion collisions. Nucl. Tech. 46, 040001 (2023). doi:10.11889/j.0253-3219.2023.hjs.46.040001

[2] Q. Chen, G.L. Ma, J.H. Chen, Transport model study of conserved charge fluctuations and QCD phase transition in heavy-ion collisions. Nucl. Tech. 46, 040013-040013 (2023). doi:10.11889/j.0253-3219.2023.hjs.46.040013

[3] K. Wang, H.S. Song, F. Fang et al., Anti-coincidence detector unit based on WLS fiber and SiPM readout. Nucl. Tech. 46, 070401 (2023). doi:10.11889/j.0253-3219.2023.hjs.46.070401

[4] H. Takeda, Y.K. Tanaka, H.Suzuki et al., Exotic nuclei and their separation, ed. by Isao.Tanihata, Hiroshi.Toki (Springer, Singapore, 2020), pp.1–59. https://doi.org/10.1007/978-981-

[5] M. Hausmann, A.M. Aaron, A.M. Amthor et al., Design of the Advanced Rare Isotope Separator ARIS at FRIB. Nucl. Instrum. Methods Phys. Res.B 317, 349–353 (2013). https://doi.org/10.1016/j.nimb.2013.06.042

[6] E. Haettner, B. Franczak, H. Geissel et al., New high-resolution and high-transmission modes of the FRS for FAIR phase-0 experi-

ments. Nucl. Instrum. Methods Phys. Res. B 463, 455–459 (2020). https://doi.org/10.1016/j.nimb.2019.04.025

[7] Y. Yang, Y.W. Su, W.Y. Li et al., Evaluation of radiation environment fragment separator HFRS at HIAF. Nucl. Sci. Tech. 29, 147 (2018). https://doi.org/10.1007/s41365-018-0479-9 in the target area of

[8] J.C. Yang, J.W. Xia, G.Q. Xiao et al., High Intensity Heavy Ion Accelerator Facility (HIAF) in China. Nucl. Instrum. Methods Phys. Res. B 317, 263–265 (2013). https://doi.org/10.1016/j.nimb.2013.08.046

[9] L.N. Sheng, X.H. Zhang, J.Q. Zhang et al., Ion-optical design of High Energy Fragment Separator (HFRS) at HIAF. Nucl. Instrum. Methods Phys. Res. B 469, 1–9 (2022). https://doi.org/10.1016/j.nimb.2020.02.026

[10] X.H. Zhou, J.C. Yang and the HIAF project team, Status of the high-intensity heavy-ion accelerator facility in China. AAPPS Bulletin 32, 35 (2022). http://dx.doi.org/10.1007/s43673-022-

[11] D. Attie, TPC review. Nucl. Instrum. Methods Phys. Res. A 598, 82–93 (2009). https://doi.org/10.1016/j.nima.2008.08.114

[12] A. Delbart, Development of resistive micromegas TPCS for T2K experiments. Nucl. Instrum. Methods Phys. Res. A 958, 162-798 (2020). https://doi.org/10.1016/j.nima.2019.162798

[13] K. Ackermann, S. Arai, A study with a small prototype TPC for the international linear collider experiment. Nucl. Instrum. Methods Phys. Res. A 623, 141-143 (2010). https://doi.org/10.1016/j.nima.2010.02.175

[14] H.K. Wu, Y.J. Wang, Y.M. Wang et al., Machine learning method for 12C event classification and reconstruction in the active target time-projection chamber. Nucl. Instrum. Methods Phys. Res. A 1055, 168528 (2023). https://doi.org/10.1016/j.nima.2023.168528

[15] The ALICE Collaboration, the CERN LHC. The ALICE Experiment Instrum. 3, S08002 (2008). https://dx.doi.org/10.1088/1748-0221/3/08/S08002

[16] W.H. Trzaska (ALICE Collaboration), New ALICE detector for Run 3 and 4 at the CERN LHC. Nucl. Instrum. Methods Phys. Res. A 958, 162116 (2020). https://doi.org/10.1016/j.nima.2019.04.070

[17] A. Di. Manro, New inner tracking system for ALICE Upgrade at the LHC. Nucl. Instrum. Methods Phys. Res. A 936, 625-629 (2019). https://doi.org/10.1016/j.nima.2018.10.047

[18] F. Shen, S. Wang. MWPC prototyping and performance tests for STAR inner TPC upgrades. Nucl. Instrum. Methods Phys. Res. A 896, 90-95 (2018). https://doi.org/10.1016/j.nima.2018.04.019

[19] K.H. Ackermann, N. Adams, STAR detector overview. Nucl. Instrum. Methods Phys. Res. A 499, 624–632 (2003). https://doi.org/10.1016/S0168-9002(02)01960-5

[20] L. Lü, H. Yi, Z. Xiao et al., Conceptual design of the HIRFL-CSR external-target experiment. Sci. China Phys. Mech. Astron. 60, 012021 (2017). https://doi.org/10.1007/s11433-016-0342-x

[21] W. Huang, F. Lu, H. Li et al., Laser test of the prototype of CEE time projection chamber. Nucl. Sci. Tech. 29, 41 (2018). https://doi.org/10.1007/s41365-018-0382-4

[22] F. García, T. Grahn, J. Hoffmann et al., A GEM-TPC in twin configuration for the Super-FRS tracking of heavy ions at FAIR. Nucl. Instrum. Methods Phys. Res. A 884, 18–24 (2018). https://doi.org/10.1016/j.nima.2017.11.088.
[23] S. Anvar et al., AGET, the GET front-end ASIC, for the readout of the Time Projection Chambers used in nuclear physics experiments. IEEE Nuclear Science Symposium Conference Record. Valencia. Spain. 745-749 (2011). doi:10.1109/NSS-MIC.2011.6154095
[24] T.X. Wang, Research and Implementation of the DAQ System for CEE. Dissertation, University of Science and Technology of China, 2022. (in Chinese)
[25] H.L. Wang, Z. Wang, C.S. Gao et al., Design and tests of the prototype beam monitor of the CSR external target experiment. Nucl. Sci. Tech. 33, 36 (2022). https://doi.org/10.1007/s41365-
[26] X. Gao, D. Emschermann, J. Lehnert et al., Throttling strategies and optimization of streaming DAQ system in the CBM experiment. Nucl. Instrum. Methods Phys. Res. A 978, 164442 (2020). https://doi.org/10.1016/j.nima.2020.164442. the trigger-less
[27] M. Jubin, D. Erno, M. Eduardo et al., Trigger and timing distributions using the TTC-PON and GBT bridge connection in ALICE for the LHC Run 3 Upgrade. Nucl. Instrum. Methods Phys. Res. A 922, 119-133 (2019). https://doi.org/10.1016/j.nima.2018.12.076
[28] Jörn Schumacher, Improving packet processing performance in the ATLAS FELIX project: analysis and optimization of a memory-bounded algorithm. DEBS '15: Proceedings of the 9th ACM International Conference on Distributed Event-Based Systems. pp 174—180 (2015). https://doi.org/10.1145/2675743.2771824.
[29] B.L. Hou, Research on the prototype electronics of RICH PID in Super Tau-Charm Facility. Dissertation, University of Science and Technology of China, 2022. (in Chinese)
[30] B.L. Hou, L. Zhao, J.J. Qin et al., Prototype of the readout electronics for the RICH PID detector in the STCF. Nucl. Sci. Tech. 33, 80 (2022). https://doi.10.1007/s41365-022-01056-4
[31] Z.H. Jia, Y. Qian, T.L. Pu et al., The design of multi-channel low noise preamplifier. Nucl. Electron. Detect. Technol. 38, 65–69 (2018).
[32] T.L. Pu, The design and research on key technology of front-end readout ASIC chips for gas detector. University of Chinese Academy of Sciences (Institute of Modern Physics, Chinese Academy of Sciences), 2020 (in Chinese).
[33] ADS52J90 Data Sheet. https://www.ti.com/lit/ds/symlink/ads-52j90.pdf. Accessed 18 Apr 2018
[34] Kintex-7 FPGAs Data Sheet. https://docs.xilinx.com/v/u/en-US/ds182 Kintex 7 Data Sheet. Accessed 26 Mar 2021
[35] Kintex UltraScale FPGAs Data Sheet. https://docs.xilinx.com/r/en-US/ug571-ultrascale-selectio/UltraScale-Architecture-SelectIO-Resources-User-Guide. Accessed 22 Sept 2020
[36] L. Zhang, J. Yang, T. Wang et al., Event building algorithm in a distributed stream processing data acquisition platform: D-Matrix. IEEE Trans. Nucl. Sci. 70, 105-112 (2023). https://doi.org/10.1109/TNS.2023.3235904

[37] W.W. Ding, C.H. Hu, Y. Cao et al., Implementation of digital constant fraction timing based on field programmable gate array. J. Phys. Conf. Series 1941, 012068 (2021). https://doi.org/doi:10.1088/1742-6596/1941/1/012068
[38] X.Q. Zhang, J.B. Fang, C.S. Lin et al., Complexity analysis of dichotomy method in solving the approximated solution of equations. IEEE International Symposium on IT in Medicine and Education, Guangzhou, pp 257-259, 2011. https://doi:10.1109/ITiME.2011.6130828
[39] G. Romeo, Elements of Numerical Mathematical Economics with Excel. Academic Press, 2020, pp 763–795. https://doi.org/10.1016/B978-0-12-817648-1.00014-1
[40] 33522B Waveform Generator, 30 MHz, 2-Channel with Arb. https://www.keysight.com/us/en/product/335
generator-30-mhz-2-channel-arb.html. Accessed 25 July 2023.
[41] J. Fang, C. Huang, T. Tang et al., Parallel programming models for heterogeneous many-cores: a comprehensive survey. CCF Trans. HPC 2, 382–400 (2020). https://doi.org/10.1007/s42514-020-00039-4
[42] G. Michael, K. Ben, A hybrid CPU/GPU approach for optimizing sorting throughput. Parallel Computing 85, 45–55 (2019). https://doi.org/10.1016/j.parco.2019.01.004

Note: Figure translations are in progress. See original paper for figures.