Glasgow College, UESTC

Project Specifications and Preliminary Report on UESTC4006P(BEng) Final Year Project

Please start by saving this file with the name:
GUID_Surname_UESTC4006P_Preliminary_year

**** ***Please add appropriate course code***

Student Name	He WenXiao	Placement Company (if appropriate)	
Student Matriculation Number	2839915H	Working Title of Project	Joint communication and control systems design
UESTC Student Number	2022190504031	Name of First Supervisor	Olaoluwa Popoola
Degree programme	Final Year Project	**Declaration of Originality and Submission Information**	*I affirm that this submission is all my own work in accordance with the University of Glasgow and UESTC Regulations, and the James Watt School of Engineering requirements.* Signed (Student): He WenXiao
Academic year	Year 4		

Your report should be NO more than 8 pages in length and include the below subject headings and incorporated within this document:

Project Description (no more than half a page)

Measurable Outcomes (no more than half a page): including Main Tasks and Targets, and Tangible outcomes (Hardware, Software, Hardware & Software, Theoretical research)

Technical Background (at least four pages)**:** including Literature Review, Topic Basis and Significance, and Research Status and Development Trend

References

Work Plan (no more than one page): including Project Outline

Resources: Complete the component request form and email the form to your 1st supervisor separately.

Rist Assessment Form: You must submit a Risk Assessment Form. Please have it signed with your 1st supervisor.

Deadlines for submission of this report: Please upload this report via the Moodle page and GC-UESTC FYP management system by the deadline mentioned in Table 1 of your project handbook.

1. Project Description

This project focuses on analyze how Joint Communication and Control (JCC) can contribute to the steady operation of modern distributed systems. JCC, which is a co-design framework, can help the autonomous agents communicate efficiently and maintain collective performance. In this project, I will be working on information compression techniques for control data exchange and explore data compression methods that maintain control accuracy while reducing communication load. This study will potentially increase the speed and quality of compression algorithm, increasing the efficiency of JCC.

2. Measurable Outcomes

2.1. Main Tasks

1. Learn the basic principles of existing algorithms
2. Conduct simulation of JCC operation and exam existing algorithms
3. Analysis potential optimizations
4. Implement and conduct experiments on potential improvement

2.2. Targets

1. Improved compression rate compared to existing algorithms
2. Increased compression speed compared to existing algorithms
3. Reduced computation consuming compared to existing algorithms

2.3. Tangible Outcomes

1. A JCC management platform that utilizes the proposed compression method
2. A demonstration in industrial application that consisting of agents connected by JCC with optimized compression algorithm

3. Technical Background

Limited bandwidth, intermittent connectivity, latency, and energy restrictions are major constraints of networked control systems and multi-agent systems. These limitations make it essential to compress control-relevant data without degrading relevant performance. This section surveys the theoretical foundations, principal compression algorithm and realistic challenges in distributed control and joint communication-control (JCC) design.

3.1. Theoretical foundations: information, rate and control performance

The control performance and the amount of communicated information is strongly connected. Early work—often called the data-rate theorem—framing stabilizability as a function of information rate. Tools based on information-theoretic tools such as rate-distortion theory have been adapted to pose trade-offs between bit rate and control cost (e.g., quadratic cost or tracking error).

3.2. Classic discrete methods: quantization and entropy coding [1]

The most direct compression tool in control systems is quantization. Uniform and nonuniform quantizers, dynamic/adaptive quantization schemes, and logarithmic scalar quantizers have been explored for state and control signal compression. Controllers and observers can be designed to tolerate quantization error through robust control techniques, state bounds, and occasional resynchronization. Entropy coding such as Huffman, arithmetic coding can be combined with quantization to exploit distributional structure and reduce average bit rate further. Key pratical considerations include encoder-decoder synchronization, bounded round-off error, and packetization for variable—length codes.

3.3. Event-triggered and self-triggered communication

Event-triggered control (ETC) shifts from regular sampling to sending updates only when a specific condition, such as an estimation or tracking error surpassing a threshold, is met. This reduces communication frequency while maintaining performance, effectively compressing data transmission. Self-triggered schemes precalculate the next transmission time, eliminating the need for constant monitoring. ETC can integrate with quantization and prioritized messaging, allocating more bits to critical updates. Research includes stability analyses under various triggering rules, robustness against delays and packet losses, and techniques for tuning triggers to balance communication rate and performance.

3.4. Sparse representations and compressed sensing [2]

Compressed sensing and sparse modeling exploit the fact that many control signals (or their derivatives/representations) are compressible in an appropriate basis. By transmitting a small number of random projections or sparse coefficient indices, the receving node can reconstruct a close approximate of the state or control vector. In distributed settings, sparsity can be enforced at the agent level (sparse control actions) or in a transform domain. Compressed sensing techniques are attractive for high-dimensional systems and sensor networks, but require careful design of measurement matrices, reconstruction algorithms (often iterative), and analysis of reconstruction error impact on closed-loop stability.

3.5. Task-oriented and control-aware compression [3]

Rather than minimizing conventional signal distortion, task-oriented compression explicitly optimizes the representation for the downstream control objective. Examples include minimizing closed-loop cost or estimation error directly in the compression loss function. This perspective leads to rate–cost optimization problems where the “distortion” is replaced by control loss. Control-aware quantizers, prioritized encoding of control-critical state components, and selective fidelity allocation across state dimensions are practical instantiations. These methods can significantly outperform generic source compression when the control task imposes structure on what information is essential.

3.6. Learning-based and end-to-end approaches

Recent trends leverage machine learning to learn compact representations tailored to control tasks. Autoencoders, variational autoencoders, and neural compressors can be trained end-to-end to generate low-dimensional messages that maintain control performance. In multi-agent systems, learned encoders capture interaction patterns and distill them into task-relevant embeddings.

Advantages: adaptability to nonlinear dynamics and complex observation models.

Drawbacks: need for extensive training data, challenges in formal guarantees, and concerns over generalization and robustness.

3.7. Joint design and co-optimization with communication

It is essential to combine compression strategies and control laws. For example, co-design of event triggers and quantizers, or optimization of control policies. These formulations allow trade-offs to be allocated optimally. Game-theoretic and other optimization methods can be used to handle decentralized decision-making in networks where agents balance local control performance and global communication resources.

3.8. Practical constraints and robustness issues

Real-world factors including variable latency, packet loss and channel noise complicate idealized compression designs. Therefore, robustness analyses must consider possible worst casesand provide bounded-error guarantees. Additionally, variable-length or probabilistic compressors complicate timing analysis in hard real-time systems. Security and privacy concerns arise when compressed messages expose sensitive state information; encryption and privacy-preserving compression introduce additional trade-offs.

3.9. Open problems and research gaps [4], [5]

1. **Bridging theory and practice**: Many existing theories about information compression and control are based on ideal or simplified models. Applying these ideas to real-world systems — which are often nonlinear, time-varying, and involve many interacting agents — is still a big challenge.

2. **Delay and packet loss**: In real wireless networks, data can be delayed or lost during transmission. Figuring out how to design compression methods that still keep the control system stable and reliable under such conditions is not yet well studied, especially for learning-based compression methods.
3. **Verifiable learning-based compressors**: Neural network compressors can achieve excellent performance, but they often lack formal guarantees of stability and safety. For safety-critical systems like drones or autonomous vehicles, being able to prove that the system will remain safe is essential.
4. **Distributed compression under coordination**: In multi-agent systems, multiple agents need to share information and work together. Designing compression protocols that allow them to communicate efficiently, protect privacy, and still coordinate effectively is an emerging and difficult research problem.
5. **Resource-aware co-design**: Real systems must balance many factors — control accuracy, energy use, computation time, delay, and bandwidth. Creating tools that can optimize all these factors together is still an open challenge, and current solutions are far from complete.

4. References

Bibliography

[1] R. Brockett and D. Liberzon, “Quantized feedback stabilization of linear systems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 7, pp. 1279–1289, 2000, doi: 10.1109/9.867021.

[2] Z. Li, Y. Xu, H. Huang, and S. Misra, “Sparse control and compressed sensing in networked switched systems,” *IET Control Theory & Applications*, vol. 10, no. 9, pp. 1078–1087, 2016, doi: https://doi.org/10.1049/iet-cta.2015.1330.

[3] A. Mostaani, T. X. Vu, S. Chatzinotas, and B. E. Ottersten, “Task-Oriented Data Compression for Multi-Agent

Communications Over Bit-Budgeted Channels," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1867–1886, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:252883663

[4] J. A. Zhang *et al.*, "An Overview of Signal Processing Techniques for Joint Communication and Radar Sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 6, pp. 1295–1315, 2021, doi: 10.1109/JSTSP.2021.3113120.

[5] H. Zhao, S. Deng, Z. Liu, J. Yin, and S. Dustdar, "Distributed Redundant Placement for Microservice-based Applications at the Edge," *IEEE Transactions on Services Computing*, vol. 15, no. 3, pp. 1732–1745, 2022, doi: 10.1109/TSC.2020.3013600.

5. Work Plan

5.1. October,2025

Select and analysis the topic and finish the primitive report

5.2. November,2025

Analyze relative papers and learn deeply about the relevant study.

5.3. December,2025

Attempt to optimize existing methods and manage to achieve satisfying results

5.4. January,2025–June 2025

Writing project report while implement my theoretical research

6. Resources

1. Software systems that can simulate industrial production with multi agents where Joint Communication and Control (JCC) is required.
2. High Performance Computation servers which can accelerate the simulation speed and handling capacity

3. A physical virtualization environment which serve as a demonstration of my proposed compression method

7. Risk Assessment Form

Num	Hazard	Possible Harm	Risk Level	Control/Action
1	**System Overcharge or Server Crash**	Data loss or experiment interruption	Medium	Regularly back up experimental data and monitor system's status
2	**Network Delay or Communication Failure**	Agent disconnection or degraded control performance	High	mplement fault-tolerant communication, use local buffering, and enable automatic reconnection
3	**Software or Configuration Error**	Simulation instability or inaccurate results.	Medium	Apply version control (Git), perform parameter verification, and conduct unit testing

Table 2: Risk Assessment Form