DECENTRALIZED NETWORKED CONTROL SYSTEMS WITH COMMUNICATION CONSTRAINTS

BY

FAISAL AHMAD AL-NASSER

A Dissertation Presented to the
FACULTY OF THE COLLEGE OF GRADUATE STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

In
SYSTEM ENGINEERING DEPARTMENT
KING FAHD UNIVERSITY OF
PETROLEUM & MINERALS

Dhahran, Saudi Arabia
OCTOBER 2012

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **FAISAL AHMED AL-NASSER** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **DOCTOR OF PHILOSOPHY IN SYSTEMS ENGINEERING DEPARTMENT.**

Dissertation Committee

Dr. Magdi S. Mahmoud (Adviser)

Dr.Fouad M. AL-Sunni (Co-adviser)

Dr. Mustafa El-Shafie (Member)

20/11/2012

Dr. Abdulwheid Al-Saif (Member)

Dr. Lahouri Chded (Member)

Dr. Fouad M. AL-Sunni
Department Chairman

Dr. Salam A. Zummo
Dean of Graduate Studies

26/11/12

Date

I would like to dedicate this thesis to my lovely Parents, My Wife and My Family ...

ACKNOWLEDGMENTS

All praise is for ALLAH (SWT), the most compassionate, the most merciful. May peace and blessings be upon prophet Mohammed (PBUH), his family and companions. I thank almighty ALLAH (SWT) for giving me the knowledge and patience to complete this work.

My deep and heartfelt gratitude and appreciation to my parents who have always sacrificed their own welfare for mine, and to whom I can only offer my unending love. It is only thanks to their prayers and blessings that I was able to complete this work. Thanks to all my family members whose love, support, encouragement and belief in me have been invaluable. Special thanks to my wife for her patience and support of me during my study period.

I would like to express my deepest thanks and appreciation to my thesis advisor Prof. Magdi S. Mahmoud, for his consistent help, guidance and attention that he devoted throughout the course of this work. His valuable suggestions and useful discussions made this work interesting to me. His support and words of encouragement gave a new life to my efforts in hard times. Without his help and valuable directions, this work would have never seen the light of the day. Thanks are also due to my thesis committee members Prof. Fouad Al-Sunni, Prof.

Mustafa Al-Shafei, Dr. Abdulwhaid Al-Saif and Dr. Lahouari Cheded for their interest, cooperation and insightful feedback.

Also, I would like to thank the Deanship of Scientific Research (DSR) at KFUPM for their support through research group project RG 1105-1.

Great thanks to Mr. Sultan Al-Dhaher, Solb Steel General Manager for his support to me and for Mr. Ameer Hijazi for his help in some programming issues. I am also thankful to all Systems Engineering faculty and staff members for their kind support and cooperation. The support and facilities provided by King Fahd University of Petroleum and Minerals for completing this work are highly appreciated.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

- AHS: Automated Highway Systems.
- AVS: Autonomous Vehicle systems.
- ATMS: Air Traffic Management Systems.
- AODV: Ad-hoc On Demand Distance Vector.
- ACS-Lite: Adaptive Control Software Lite.
- CAN: Controller Area Network.
- CSMA/CA: Carrier Sense Multiple Access / Collusion Avoidance.
- DNCS: Distributed Networked Control System.
- DecNCS: Decentralized Networked Control System.
- DSDV: Destination Sequenced Distance Vector.
- DSR: Dynamic Source Routing.
- ENN: Extension Neural Network.
- EDF: Earliest Deadline First.
- FP: Fixed Priority.
- GCNC: Guaranteed Cost Networked Control.
- HOV: High Occupancy Vehicle.

- LMI: Linear Matrix Inequality.
- LSS: Large-Scale System.
- LQ: Linear Quadratic.
- LTI: Linear Time Invariant.
- LTV: Linear Time Variant.
- LQR: Linear Quadratic Regulator.
- LQG: Linear Quadratic Gaussian.
- MAC: Medium Access Protocol.
- MHS: Material Handling Systems.
- MADB: Maximum Allow Delay Bounded.
- MIMO: Multi Inputs Multi Output.
- MPC: Model Predictive Control.
- NCS: Networked Control System.
- NEMA: The National Electrical Manufacturers Association.
- PID: Proportional Integral Derivative.
- OPAC: Optimized Policies for Adaptive Control.
- QCS: Quantized Control System.

- QuasiNCS: Quasi-Decentralized Networked Control System.
- QoS: Quality of Service.
- RTO: Real Time Optimization.
- RM: Rate Monotonic.
- RL: Reinforcement Learning.
- RHODES: Real Time Hierarchical Optimized Distributed Effective System.
- SCOOT: Split Cycle Offset Optimization Technique.
- SCATS: Sydney Coordinated Adaptive Traffic System.
- TCS: Traffic Control System.
- TCP/IP : Transmission Control Protocol/ Internet Protocol.
- UDP: User Datagram Protocol.
- WEP: Wired Equivalent Privacy.
- WNCS: Wireless Networked Control System.
- WSN: Wireless Sensor Network.
- ZOH: Zero-Order Hold.
- PID: Proportional Integral Derivative Controller.
- PLC: Programmable Logic Controller.
- MPC: Model Predictive Control.

DISSERTATION ABSTRACT

NAME:	Faisal Ahmed Al-Nasser
TITLE OF STUDY:	Decentralized Networked Control Systems with Communication Constraints
MAJOR FIELD:	Systems Engineering Department
DATE OF DEGREE:	October 2012

In this work, we investigate a decentralized control approach to dynamical systems where the control loops are spread over a network. In such a decentralized networked control systems (DecNCS), the subsystems are communicating with each other via a shared communication network. The properties of the network such as delay, packet dropout, varying sample interval and induced errors must be studied when designing the control system. These properties add restrictions and difficulties in the control loop that are not present in traditional control loops. It is important in the design of the control strategy to identify how the control loop is structured and where the network comes into the picture. Also, the general structure of a decentralized networked control system is described along with the main characteristics, major problems, network communication parameters and the tech-

niques of handling lost control data. The proposed approach that is applied to a selected real-life application, which is the signalized traffic coordination and control problem, is discussed here over different traffic light control structures over communication links, including the decentralized, quasi-decentralized, distributed and hierarchical networked structures. These structures are used for coordinating multiple intersections, which could be a great application of networked control problem control for the signalized traffic light intersections that will help the designer to achieve certain objectives. Some of these objectives are to minimize the waiting time during the red light period and perform better control in the next green cycle, maximize the flow between consecutive intersections which will minimize both the number of stops and the average waiting time during the trip. Other objectives will also be highlighted in this work. An extensive and collective literature survey about all models used for traffic control problem is presented here. A state-space model of traffic dynamics under these different control structures is proposed. The model takes into account the effects of lossy communication links such as networked induced delays, packet dropout, communication constraints and varying sample interval. Also, a sufficient condition for system stability is provided based on LMI. Finally, a comparison of different types of networked control systems and performance analysis were done using simulation.

ملخص بحث

درجة الدكتوراة في الفلسفة

الاسم: **فيصل أحمد الناصر**

العنوان: أنظمة التحكم اللامركزية عبر شبكات الاتصال ذات الإمكانات المحدودة والمقيدة.

التخصص: هندسة نظم التحكم والقياس

تاريخ الشهادة: **أكتوبر 2012 م**

في هذه الأطروحة ،تتم دراسة أنظمة التحكم اللامركزية لمراقبة الأنظمة الديناميكية حيث تنتشر حلقات التحكم على الشبكة . في مثل هذه النظم اللامركزية تتم عمليات التحكم وتتواصل النظم والمتحكمات مع بعضها البعض عبر شبكة الاتصال المشتركة ذات الإمكانات المحدودة والمقيدة. لذلك يجب أن يتم دراسة تأثير مثل هذه الشبكة على نظام التحكم مثل تأخير وصول المعلومات وأوامر التحكم، فقدان حزم التحكم، فترات الفاصل الزمني المتغيرة لأخذ القراءات والأخطاء التي تسببها القراءات المتفاوتة. هذه الأمور تؤدي إضافة قيود وصعوبات أثناء عمليات تصميم حلقة التحكم التي قد لا تواجهنا في تصميم حلقات التحكم التقليدية وقد تكون لها آثار سلبية قد تعرض استقرار النظام للخطر أو تسبب رداءة في الأداء. من المهم في تصميم استراتيجية التحكم تحديد كيفية بناء حلقة التحكم و السيطرة، وأين يأتي دور شبكة الاتصال المشتركة في التصميم. يتضمن هذا العمل أيضا وصف الهيكل العام للنظام اللامركزي عبر الشبكات المقيدة الاتصال جنبا إلى جنب مع الخصائص والمشاكل الرئيسية لمثل هذه الأنظمة، وأساليب التعامل مع البيانات والإشارات المفقودة. بعد ذلك تم عمل المحاكاة لهذه الأنظمة التي تمت دراستها وتحقيق صحتها على أنظمة قياسية معروفة الخصائص.

بعد ذلك تم تطبيق نتائج الدراسة على تطبيقات من واقع الحياة المعاصرة، وقد تم اختيار تنسيق و تنظيم حركة المرور بين تقاطعات الطرق ذات الإشارات الضوئية ، وقد تمت مناقشة مختلف هياكل مراقبة الحركة المرورية المحكومة بالإشارات الضوئية ذات الاتصال، بما في ذلك اللامركزية والموزعة والهرمية . وتستخدم هذه الهياكل لتنسيق الحركة المرورية بين التقاطعات ذات الإشارات الضوئية والتي يمكن أن تعتبر تطبيقا كبيرا في مجال السيطرة والتحكم بالشبكة ذات التقاطعات المرورية المتتابعة والتي تساعد مهندس التحكم والمرور لتحقيق أهداف معينة. هنالك الكثير من الأهداف التي يمكن تحقيقها والتي منها على سبيل المثال تقليل وقت الانتظار خلال انتظار إشارة الضوء الأحمر وأداء تحكم أفضل في دورة الإشارة الخضراء التالية، وزيادة لإمكانية تدفق المركبات وسيرها بشكل متواصل بين التقاطعات على التوالي دون الحاجة للتوقف عند كل إشارة ضوئية وهو الأمر الذي يقلل من وقت الانتظار خلال الرحلة وبالتالي إجمالي الوقت اللازم لقطع المسافة بين هذه التقاطعات. كما سيتم تسليط

الضوء على أهداف أخرى في هذا العمل ودراستها. وتقدم الأطروحة دراسة استقصائية وبحثية واسعة النطاق حول عدد كبير من النماذج المستخدمة لمراقبة وتنظيم الحركة المرورية.

أيضا تم تمثيل هذه المسائل بنماذج رياضية اعتمادا على طرق معينة تشمل أنظمة التحكم المختلفة حيث تأخذ هذه النماذج في الاعتبار آثار روابط الاتصال مع الفقدان الناجم عن التأخير، وتسرب الحزم، والقيود على الاتصالات وتفاوت الفاصل الزمني للقراءات. وقد تم اعتماد شروط كافية لاستقرار النظام على شكل متراجحات مصفوفية خطية لعدد من مفاهيم الإستقرار. وبعد ذلك تم القيام بمقارنة أنواع مختلفة من نظم التحكم عبر الشبكة التي تمت منقاشتها وتحليل الأداء لكل منها باستخدام المحاكاة ثم مناقشة الصعوبات الناتجة عن كل نظام . وفي ختام العمل تم طرح بعض النقاط التي يمكن إضافتها إلى هذا العمل مستقبلا .

CHAPTER 1

INTRODUCTION

1.1 Motivation

Centralized control, although very widely deployed , is neither robust nor scalable to complex large-scale dynamical systems with their measurements distributed over a large geographical region. The main reasons for this, are first, the computational complexity of employing such centralized controller is very high. Second, the distribution of the sensors over a vast geographical region poses a large communication burden which may add long delays and loss of data to the control process. Third, the centralized mechanism is harder to adapt to the changes in the large-scale system. Fourth, the large-scale system can be composed of smaller subsystems with poorly modeled interactions between them and the centralized control is not robust to such interactions.

Decentralized Control offers a classical alternative which removes the difficulties caused by centralization. In this approach, the large-scale system is decomposed into N subsystems. This decomposition can be constructed based on the geographical distribution of the global system, constraints on the measurements availability, weak coupling between the subsystems, and many other criteria. After the system decomposition, a local low-order control is built for each subsystem so that it operates on local measurements. Hence, decentralized control of large-scale systems is having an increasingly important role in real-world problems because of its scalability, robustness and computational efficiency. Applications can be found in many areas like aircraft formations, power systems, platoons control, environmental monitoring, water transportation networks and communication networks.

In the other hand, the recent technological advances in communication networks and the decreasing in cost and size of electronics have promoted the appearance of large inexpensive interconnected systems, each with computational and sensing capabilities. Therefore, the systems are distributed with components communicating over networks. Such a setting has a number of features that seriously challenge the controller design. First, the controllers are in decentralized structure which means that they do not share information. If we want to consider a centralized controller, then huge bandwidth associated with using a centralized control structure would be limited by long delays induced by the communication between the centralized controller and distant sensors and actuators over a communication network. The second challenge, when considering control of a large-scale system, introduce the unpractical assumption that all states are measured. Therefore an output-based controller is needed.

Note that an observer-based controller offers the advantage of reducing the number of sensors, which alleviates the demands on the network design. Finally, the observer-based controller needs to have certain robustness properties when using a communication network. However, the drawback is that the control system is susceptible to undesirable side-effects such as: time-varying delays, packet dropouts, varying sampling intervals, quantization and communication constraints (the latter meaning that not all information can be sent over the network at once). Other approaches are also presented to overcome the problems with decentralized control which is the quasi-decentralized where we allow limited communications which is a very useful approach and need lower bandwidth than the distributed control where all communications are allowed.

After reviewing a number of architectures for the control of interconnected and networked control systems considering a widespread industrial application of distributed, networked and hierarchical solutions, many fundamental problems are yet to be solved. Moreover, Many theoretical contributions are required to develop efficient algorithms with guaranteed properties, such as stability and performance for decentralized networked control systems, where few results are available for the decentralized control over a network.

1.2 Organization of the thesis

This thesis contains five chapters followed by appendices that cover some important control knowledge. In the following we will describe in brief each chapter:

- Chapter 1 which is the introduction.

- Chapter 2 provides a collective survey about different control structures of decentralized, distributed, hierarchical and networked control for interconnected dynamical systems. Attention is focused on the classification of control approaches and also presents the control extensions for each.

- Chapter 3 investigates a decentralized control approach to dynamical systems where the control loops are spread over a network. In such a decentralized networked control systems (DecNCS), the subsystems are communicating with each other over a shared communication network. The general structure of a decentralized networked control system is described, along with the main characteristics, major problems, network communication effects and the techniques of handling lost control data. Also, we have presented a generic model for a discrete-time linear system that captures all network aspects and provides insight into how they influence each other. A Closed-loop model is derived based on LMI stability conditions. Finally, simulation of standard system and solution verification were presented.

- Chapter 4, This chapter starts with motivation of improving performance by coordinating multiple intersections based on different traffic light control structures over communication links, including the decentralized, quasi-decentralized, distributed and hierarchical networked control. These structures are used for signalized traffic multi-intersections control and coordinating, which could be a great application of networked control problem in the field of the traffic engineering and control. It will also help the designer or the control engineer to achieve certain objectives. Some of these objectives are to minimize the waiting time during the red light period and perform better control in the next green cycle, maximize the flow between consecutive intersections which will minimize the number of stops, minimize the average waiting time during the trip and more will be highlighted in this chapter. An extensive literature survey was done about all models used for traffic control problem. A state-space model of traffic dynamics under different control structures is proposed. The model takes into account the effects of lossy communication links such as networked induced delays, packet dropout, communication constraints and varying sample intervals. Also, a sufficient condition for system stability is provided based on LMI. Finally, detailed simulations for the selected application of different experiments were done on multi signalized intersections control and coordination with different environments, control strategies and targets (e.g. showing the effects of network, performance comparison between proposed control strategies, complexity issues...etc).

- Chapter 5 include the conclusion and possible future work directions.

1.3 Thesis Objectives

This thesis covered several objectives and all were achieved and to the best of our knowledge, the problems solved were not dealt within the literature before in the way it was treated here. The following shows the main objectives and contributions of the thesis:

1. Comprehensive Literature survey on control strategies.
2. Develop a generic model describing the decentralized, quasi, distributed control systems over a communication network considering all network side effects.
3. An observer based controller design that is robust to the communication link side effects and the related closed Loop models for each control strategy.
4. Develop a sufficient condition for system stability (LMI-based).
5. Compare different types of networked control systems using simple simulation.
6. Comparative Study applied on control application (Signalized Traffic Multi-Intersections Control (STMIC)) with an extensive and collective survey on the selected application.
7. Develop a state space model for the signalized intersection control problem and traffic dynamics that covers the proposed control strategies (Decentralized, Distributed and Quasi-Decentralized) and it must take into account the effects of lossy communication links.

8. Present a meaningful discussion on traffic constraints,complexity study and cycle time.

Also, we will list the publications out of this thesis and the submitted ones.

1. Paper submitted: Title **"New Approach to Decentralized Control over Communication Networks"** to the *International Journal of Robust and Nonlinear Control.*
2. Paper submitted Title **"Control Strategies over Communication Network for Signalized Traffic Intersections"** to the *IEEE Transactions on Vehicular Technology.*
3. Paper submitted: Title **Architectures for Distributed and Hierarchical Networked Control Systems- A Survey** , Int. J. Systems Science.
4. Paper submitted and accepted: Title **"Signalized Traffic Intersections Control with Uncertainties Over Lossy Networks"**, to the 4th International Conference of Soft Computing and Pattern Recognition, SoCPaR 2012.
5. Book Chapter Published: Title Book title: **Wireless Sensor Networks -Technology and Applications** and the Chapter title: **Wireless Sensors Network Applications:A Decentralized Approach for Traffic Control and Management**, **InTech**, ISBN 978-953-51-0676-0, Published in July 18, 2012.

1.4 Terms and Terminology

In the following, we define some terms that are used in the forthcoming chapters of this thesis.

1. Controller: In control theory, a controller is a device, possibly in the form of a chip, analogue electronics, or computer, which monitors and physically alters the operating conditions of a given dynamical system to achieve a desired objective.

2. Sensor: A sensor (also called detector) is a converter that measures a physical quantity and converts it into a signal which can be read by an observer or by an (today mostly electronic) instrument. (e.g. A thermocouple converts temperature to an output voltage which can be read by a voltmeter).

3. Actuator: It is a device which transforms an input signal (e.g. an electrical signal) into action (e.g. motion). Electrical motors, pneumatic actuators, hydraulic pistons, relays, comb drives, piezo-electric actuators and electroactive polymers are some examples of such actuators.

4. Open Loop Controller: An open-loop controller, also called a non-feedback controller, is a type of controller that computes its input into a system using only the current state and its model of the system.

5. Closed-Loop Controller: Also, called Feedback Controller, is a process in which information about the past or the present influences the same phenomenon in the present or future output. As part of a chain of cause-and-effect that forms a circuit or loop, the event is said to "fed back" into itself.

6. Nash Equilibrium: A concept of game theory where the optimal outcome of a game is one where no player has an incentive to deviate from his or her chosen strategy after considering an opponent's choice. Overall, an individual can receive no incremental benefit from changing actions, assuming other players remain constant in their strategies. A game may have multiple Nash equilibria or none at all.

7. Queue/Link Capacity: is defined by the maximum number of vehicles for link i and it can be determined by the length of link between two intersections.

8. Cycle: A cycle is a complete sequence of intervals or a complete sequence of signal indications.

9. Cycle Time: the cycle time Tc is the time required to complete the execution of all phases for the intersection and it shall be bounded by a certain maximum value.

10. Cycle offset: defines the starting time of a cycle relative to other traffic lights and it can be adjusted to let several lights cooperate and lead to green waves.

11. Green Period: or green time Tg is the interval in seconds of green indication for link or queue at a signalized intersection i shall not exceed certain value to be fair with other links during the same cycle time.

12. Phase: a phase is any period in a cycle where non-conflicting traffic movements may run. A phase is the part of the cycle assigned to a fixed set of traffic movements. When any of these movements change, the phase changes.

13. Yellow Change interval: This is an interval in which yellow indications tell drivers in the phase with the right-of-way that their movement is about to lose its right-of-way.

14. Red Clearance interval: This describes the interval when all of the indications are red and is a safety measure designed to give the oncoming traffic enough time to clear the intersection before the next phase begins.

15. Intergreen time: This is the summation of the time allocated to the change and clearance intervals for a given phase (yellow and all red time).
16. Green Split: how long each phase will have the right of way (green indication).
17. Effective green time: which is the time that a movement is going, regardless of the indication shown (i.e. people going on yellow or not going at the start of a phase). Also, can be defined as the time during which a given traffic movement or set of movements may proceed; it is equal to the cycle length minus the effective red time.
18. Saturation flow rate: is the number of vehicles served by a lane for one hour of green time. In order to determine saturation flow rate, we must know the headway and saturation headway. Also, can be defined as the equivalent hourly rate at which previously queued vehicles can traverse an intersection approach under prevailing conditions, assuming that the green signal is available at all times and no lost times are experienced.
19. Headway: is the time interval between the passage of successive vehicles moving in the same lane measured from head to head as they pass a point on the road.
20. Saturation headway: is the headway of the vehicles in a ”stable moving platoon” passing through a green light.
21. Change and clearance interval: The yellow plus all-red interval that occurs between phases of a traffic signal to provide for clearance of the intersection before conflicting movements are released.
22. Clearance lost time: The time between signal phases during which an intersection is not used by any traffic

23. Control delay: The component of delay that results when a control signal causes a lane group to reduce speed or to stop; it is measured by comparison with the uncontrolled condition.
24. Effective red time: The time during which a given traffic movement or set of movements is directed to stop; it is equal to the cycle length minus the effective green time.
25. Extension of effective green time: The amount of the change and clearance interval, at the end of the phase for a lane group, that is usable for movement of its vehicles.
26. Interval: A period of time in which all traffic signal indications remain constant.
27. Red time: The period in the signal cycle during which, for a given phase or lane group, the signal is red
28. Start-up lost time (Startup Delay): The additional time consumed by the first few vehicles in a queue at a signalized intersection above and beyond the saturation headway, because of the need to react to the initiation of the green phase and to accelerate.
29. Total lost time: The total lost time per cycle during which the intersection is effectively not used by any movement, which occurs during the change and clearance intervals and at the beginning of most phases.
30. NEMA : The National Electrical Manufacturers Association (NEMA) is a trade association with 450 member organizations that sets standards for the generation, distribution, transmission, control and end-use of electricity. NEMA works in conjunction with the National Transportation Commu-

nications for Intelligent Transportation Systems Protocol to set standards governing traffic signals.

31. Flow (f): Number of vehicles passing a certain point during a given time period, in vehicles per hour (veh/hr).
32. Speed (v): The rate at which vehicles travel (Km/h)
33. Density (d): Number of vehicles occupying a certain space. Given as veh/m, $d = f/v$.
34. Shockwave: Low density traffic meets high density traffic.

1.4.1 Notations and Facts

In the sequel, the Euclidean norm is used for vectors. We use W^t and W^{-1} to denote the transpose and the inverse of any square matrix W, respectively. We use $W > 0$ to denote a symmetric positive definite matrix W and I to denote the $n \times n$ identity matrix. Matrices, if their dimensions are not explicitly stated, are assumed to be compatible for algebraic operations. In symmetric block matrices or complex matrix expressions, we use the symbol $\bullet$ to represent a term that is induced by symmetry.

Sometimes, the arguments of a function will be omitted when no confusion can arise.

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

A wide class of control systems consists of interconnected controllers, sensors and actuators that have been successfully implemented using point-to-point architecture where all components are directly wired to various controllers. The goal is to design flexible systems that can accomplish various tasks with minimum configuration cost, maximum reliability while achieving the desired performance. In feedback control there are two major operations to be accomplished: the transmission of signals (information flow) back and forth and the calculation of control actions (decision making). Various control architectures can be implemented to perform these operations in a real plant: centralized, decentralized and distributed control architectures. A centralized control system utilizes one centralized controller which collects data from all sensors, computes control decisions, and then dispatches actuation signals to actuators. It is well known that the centralized control can provide the best performance since it imposes the least constraints on the control structure under consideration. However, this structure has a single point of failure [10] plus the computational and organizational complexity associated with centralized controllers often makes their implementation impractical, especially for plants with complex dynamics and/or spatially distributed in space such as the Large Scale System (LSS).

These considerations have eventually motivated significant work on decentralized control [2], where multiple sensors and controllers are placed in a distributed structure, and the control decisions are made with or without the full set of response data. The controller nodes can be closely collocated with system actuators. The major issue in this structure is due to lack of communication between controllers [34], therefore, the closed-loop performance of the plant may deteriorate, and in some cases stability may be lost. In this regard, significant research work [10, 11, 12, 52] has explored in depth the benefits and limitations of decentralized controllers as well as possible ways of overcoming some of their limitations. Another approach is the distributed control where the communication between controllers is allowed [154]. It stands nowadays as the technologically efficient infrastructure for many advanced control strategies in industry that are often implemented over local and proprietary networks.

However, the great availability and ever-decreasing costs of the networked technology have been responsible for the replacement of the traditional point-to-point link for broadcast transmission. That motivates the implementation and the usage of shared network to connect spatially distributed elements results in flexible architectures which help in reducing the installation and maintenance costs. Also, they present better characteristics in terms of modularity, scalability and offers more design options [1, 5, 66]. By closing the feedback control loop over a communication network, this will introduce the Networked Control System (NCS) [23, 26, 67, 68]in which the sensor, the actuator, and the controller are elements that share information by exchanging messages over the network.

Consequently, this type of control systems have been finding applications in a broad range of areas such as mobile sensor networks [79], remote surgery [80], haptics collaboration over the Internet [81, 82, 83] , automated highway systems, unmanned aerial vehicles, control of surveillance and rescue robot teams for access to hazardous environments and space exploration [84, 85]. In such remote control applications, one major concern is the characterization of a sufficient amount of information transfer needed for a satisfactory performance. This information transfer can be between various components of a networked control system. One necessity for satisfactory control performance is the ability for the controllers to track the plant states under communication constraints. Another challenge is the determination of the data rate required for the transmission of control signals, and the construction of dynamic encoding, decoding, and control policies meeting some criteria. One more important problem also is the coordination among multiple sensors or multiple controllers/decision makers with the lowest information exchange possible. Moreover, the presence of a network in the control loop has the drawback of introducing time delays in the communications among field elements. Depending on the network configuration, these delays can exhibit random behaviour and may even cause variation in the sampling periods, which in turn, result in a time-varying plant [68, 69]. These delays deteriorate the performance and even the stability of the system as a whole [70, 71, 72, 75].

Therefore, it is not a trivial task to design communication protocols or architectures for control systems since both communication delay and packet loss negatively impact estimation, closed-loop performance and other parameters of controlled systems [37]. Currently, communications protocols and networked control systems are designed separately. In particular, protocols are designed based on conservative heuristics which, by and large, specify what the maximum time delay and maximum packet loss should be. The delays may not affect significantly an open loop control system but for the applications which require feedback data across the network, the traditional control strategies which can deal with constant time delays might not be suitable. Moreover, real-time implementation can introduce non-constant time delay (jitter) in the control loop as the result of tasks pre-emption and priorities [76, 77]. Each control task contains three parts; sampling, control algorithm and actuation. The jitter can happen in sampling and/or actuation. The real-time scheduling can affect control system performance because of the jitter that is imposed by scheduling while the control system design can affect the real-time scheduling since the period of control task is determined during the control design stage. Hence, it is required to integrate the design of control system and real-time scheduling to eliminate the effect of jitter and achieve the desired requirements.

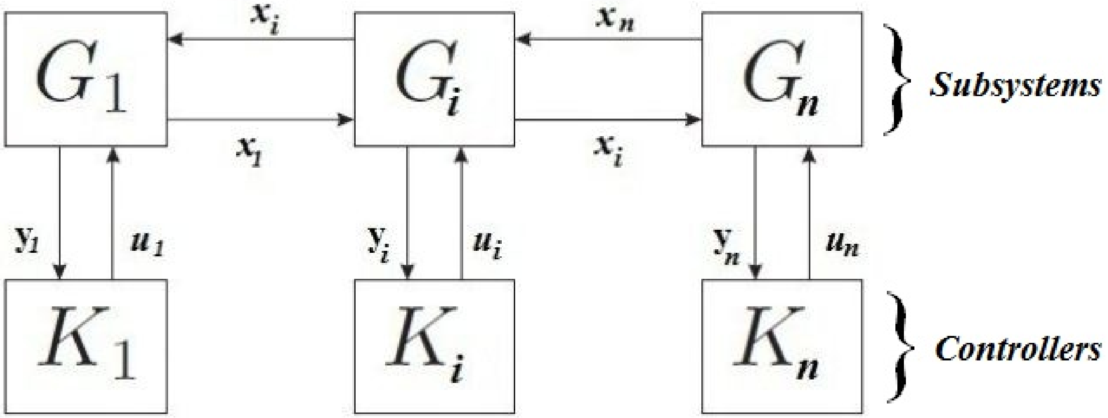
From these observations few questions arise [4]: How should we design estimators for networked systems that take into account simultaneous random delay and packet loss? How can we estimate their performance? When is the closed loop system stable? How can we choose between a communication protocol with a large packet delay and a small packet loss and a protocol with a small packet delay and a large packet loss, in terms of best performance of a specific real-time application? Indeed, the situation becomes compounded when considering interconnected dynamical systems. From this perspective, the objective of next sections is to review a number of control architectures such as decentralized, distributed, networked and hierarchical networked control and providing an overview for each type.

2.2 Decentralized Control

In this paradigm, the plant is decomposed into a number of simpler subsystems (typically based on functional and/or time-scale differences of the unit operations) with interconnections, and a number of local controllers are connected to each distributed subsystem with no signal transfer taking place between different local controllers [10]. Decentralized control of multi-unit plants can reduce complexity in the controller design and implementation, and can also provide flexibility in dealing with local controller failures. In a decentralized control system, since each controller communicates with sensors and actuators in its vicinity, the requirement on communication range can be significantly reduced, and communication latency decreases by reducing the number of sensors or actuators that each controller has to communicate with [34, 35]. As the control decisions are computed and executed distributively by individual controllers, system redundancy and reliability can also be improved using decentralized control. On the other hand, since each controller may only have limited information and longer time delay using communication medium, the stability and optimality of decentralized control strategies need to be examined carefully.

In decentralized architectures, the control (input u) and the controlled (output y) variables are grouped into disjoint sets. These sets are then coupled to produce non-overlapping pairs for which local controllers are designed to operate in a completely independent fashion. The local controllers can be single-input single-output or multivariable (locally centralized) depending on the cardinality of the selected input and output groups. An example of a perfectly decentralized control structure is shown in Fig. 2.1, which provides local posterior information to each controller with no information exchanged between local controllers. Knowledge of how the control actions of the local controller affect the overall system response is not known.

The system under control is assumed to be composed by n subsystems, with states, control and output variables (x_i, u_i, y_i), where $i = 1, ..., n$, and the interaction between the subsystems is due to the mutual effect of the states. Once the decentralized controller structure has been defined, the design of the local



$$\begin{bmatrix} u_1 \\ \cdot \\ u_i \\ \cdot \\ u_n \end{bmatrix} = \begin{bmatrix} K_1 & & & & 0 \\ & \ddots & & & \\ & & K_i & & \\ & & & \ddots & \\ 0 & & & & K_n \end{bmatrix} \begin{bmatrix} y_1 \\ \cdot \\ y_i \\ \cdot \\ y_n \end{bmatrix}$$

Figure 2.1: Decentralized Control System

controllers (K_1 ..., K_n), becomes a routine task when the interactions among the inputs and the outputs of different pairs are weak. These interactions can either be direct (input coupling) or caused by the mutual effects of the internal states of the subsystems under control. On the contrary, it is well known that strong interactions can even prevent one from achieving stability and/or performance with decentralized control.

In many practical situations, complete state measurements are not available at each individual subsystem for decentralized control; consequently, one has to consider decentralized feedback control based on measurements only or design decentralized observers to estimate the state of individual subsystems that can be used for estimated state feedback control. The general model for a decentralized linear observer that will track the plant states in the presence of bounded interconnections is shown in the following state equations:

$$\begin{aligned} \dot{\tilde{x}}_i &= A_i\tilde{x}_i + B_iu_i + L_i(y_i - C_i\tilde{x}_i) \\ \dot{\tilde{y}}_i &= C_i\tilde{x}_i, \quad i = 1,\ ...,\ n \end{aligned} \tag{2.1}$$

L_i is the observation gain matrix of i^{th} subsystem ,$\tilde{x}$ is the state observation of i^{th} subsystem. Notice that the state observation structure of the global interconnected subsystems is completely decentralized since there is no information transfer between local observers.

The dynamic of the observation error between the i^{th} true state and the i^{th} observer output is:

$$e_i = x_i - \tilde{x}_i \tag{2.2}$$

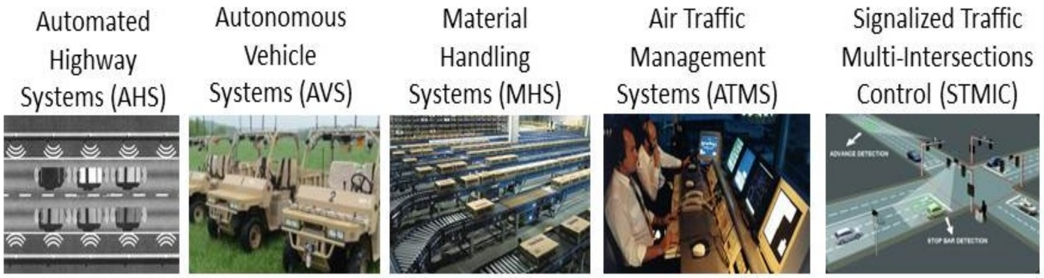
The local control law of each subsystem is given by

$$u_i = -K_i \tilde{x}_i(t) \tag{2.3}$$

More works also is available in the literature for developing different decentralized techniques such as adaptive, robust and nonlinear decentralized control [116]-[121].

2.3 Large Scale Systems (LSS)

The large scale systems (LSS) usually are physically distributed over a wide area and the decentralized controllers are the most applicable control philosophy for LSS [2]-[9],[114] and this is so because of the information exchange between subsystems of a LSS is not needed; thus, the individual subsystem controllers are simple and use only locally available information. Large-scale interconnected systems can be found in such diverse fields such as automated highway systems (AHS), autonomous vehicle systems (AVS), material handling systems (MHS), air traffic management systems (ATMS), manufacturing, power generation and distribution [115]. Designing a centralized control for these systems may not be efficient due to the natural system's modularity, which may prevent a viable way of sharing information across the subsystems, and often it may be too costly when it is implemented. These limitations motivate the design of decentralized control systems


Automated Highway Systems (AHS)

Autonomous Vehicle Systems (AVS)

Material Handling Systems (MHS)

Air Traffic Management Systems (ATMS)

Signalized Traffic Multi-Intersections Control (STMIC)

Figure 2.2: LSS Examples

where the information exchange between interconnected subsystems of large-scale systems is not required. Thus, the control law implementation is more feasible and more economical. However, in the decentralized state feedback control, the state variables are not available or are costly to measure. Therefore, it is necessary to design a state observer to reconstruct the individual states of each subsystem.

Two broad methods can be used to design observer-based decentralized output feedback controllers for large scale systems:

- Design local observer and controller for each subsystem independently and check the stability of the overall closed-loop system. In this method, the interconnection in each subsystem is regarded as an unknown input.
- Design the observer and controller by posing the output feedback stabilization problem as an optimization problem.

The aim of any control design is not only to stabilize the system but also to ensure satisfactory performance of that system. For linear systems the quadratic cost is characterized by the LQ (Linear Quadratic) design which offers an optimal solution. Whereas for nonlinear and/or uncertain systems the guaranteed cost control, in the presence of admissible nonlinearities and/or uncertainties, ensure that the closed loop system is asymptotically stable and an upper bound of the quadratic cost is minimized. Hence, the result of these control designs are characterized in terms of parameterized algebraic Riccati equations which are may be difficult to solve.

The large scale interconnected systems characterized by linear subsystems for which are added nonlinear interconnections, are modeled in [115] and [117] as the following:

$$\begin{aligned} \dot{\tilde{x}}_i &= A_i x_i + B_i u_i + h_i(t, x(t)) \\ y_i &= C_i \tilde{x}_i, \quad i = 1,\ ...,\ n \end{aligned} \tag{2.4}$$

Where x_i is the state vector of i^{th} subsystem, u_i is the control vector of i^{th} subsystem, y_i is the output vector of i^{th} subsystem and h_i reflects the interconnection term illustrating the nonlinearity of i^{th} subsystem. The matrices A_i, B_i and C_i denote respectively the state matrix, the control matrix and the output matrix of each subsystem with the following:

- (A_i, B_i) assumed to be controllable.
- (A_i, C_i) assumed to be observable.

2.4 Quasi-Decentralized Control

To solve the problem where a decentralized control structure cannot provide the required stability and performance properties, and to avoid the complexity and lack of flexibility associated with traditional centralized control, a quasi-decentralized control (partially decentralized and not fully distributed) strategy with minimum cross-communication between the plant units offers a suitable compromise and

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} K_{11} & DK_{12} & 0 & 0 & 0 \\ DK_{21} & K_{22} & DK_{23} & 0 & 0 \\ 0 & DK_{32} & K_{33} & DK_{34} & 0 \\ 0 & 0 & DK_{43} & K_{44} & DK_{45} \\ 0 & 0 & 0 & DK_{54} & K_{55} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}$$

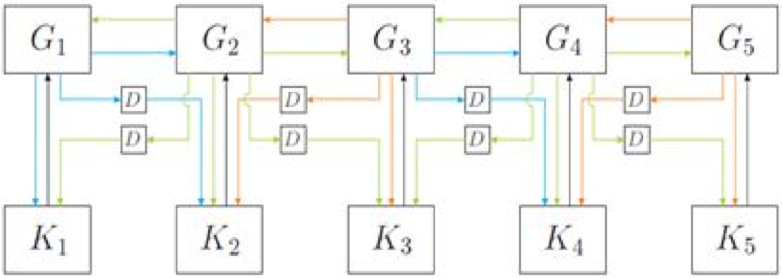


Figure 2.3: Quasi-decentralized Control System

it provides a way of ensuring partial knowledge of how the local controller is affecting the global system. Most of the signals used for control are collected and processed locally, although some signals (the total number of which is kept to a minimum) still need to be transferred between local units and controllers to adequately account for the interactions between the different units and minimize the propagation of disturbances and process upsets from one unit to another [9].

One of the key problems that need to be addressed in the design of quasi-decentralized control systems [51, 52] is how to coordinate the control and communication functions and how to account for possible limitations of the communication medium in the formulation and solution of the control problem especially if the communication is a shared medium like Ethernet. This is an important problem in view of the increased reliance in the process industries in recent years on sensor and control systems that are accessed over communication networks rather than hardwired.

The design of a quasi-decentralized control [53, 54] strategy that enforces the desired closed-loop objectives with minimal cross communication between the component subsystems is an appealing goal since it reduces reliance on the communication medium and helps save on communication costs. This is an important consideration particularly when the communication medium is a potentially unreliable (e.g. wireless sensor network) where conserving network resources is key to prolonging the service life of the network.

2.5 Distributed Control

In distributed control structures, like the simple example shown in Fig. 2.4, it is assumed that the information is transmitted among the local controllers so that each one of them has knowledge on the behaviour of the others [149]-[154], [127]. The information transmitted, for example, can consist of the future predicted control or state variables computed locally, so that any local controller can predict the interaction effects over the considered prediction horizon. A classification can be made depending on the topology of the communication network. Specifically, the following cases can be considered:

- Information is transmitted (and received) from any local controller to all the others (fully connected algorithms).
- Information is transmitted (and received) from any local controller to a given subset of the others (partially connected algorithms).

Distributed control system consisting of N subsystem discrete-time linear dynamic model of the subsystem i can be described as following:

$$x_i(k+1) = \sum_{j=1, j\neq i}^{N} (A_{ij} x_j(k)) + G_i w_i(k) \tag{2.5}$$

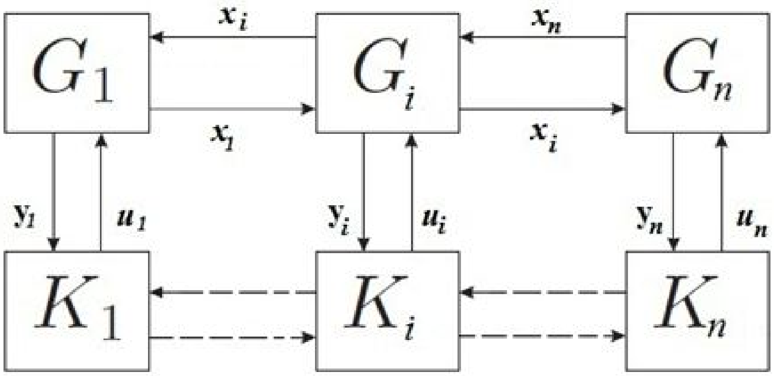
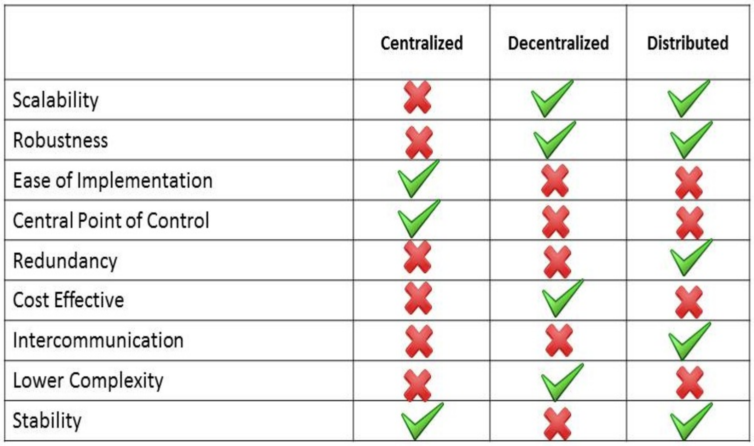


Figure 2.4: Distributed Control System

Where x_i is the state of the subsystem i at instant k, w_i is the white noise process [27].

A partially connected information structure can be convenient in the case of large-scale systems made by a great number of loosely connected subsystems. In these cases, restricting the information exchange among directly interacting subsystems produces negligible performance deterioration. An interesting discussion on this point is covered in [13], where reference is made to chemical processes composed by subsystems directly interacting only with their neighbours, possibly with additional recirculating flows. It is apparent that the amount of information available to the local controllers with iterative algorithms is higher, so that an overall iterative procedure can be set-up to reach a global consensus on the actions to be taken within the sampling interval. To this regard however, a further classification has to be considered:

- Distributed algorithms where each local controller minimizes a local performance index (independent algorithms).
- Distributed algorithms where each local controller minimizes a global cost function (cooperating algorithms).



	Centralized	Decentralized	Distributed
Scalability	✗	✓	✓
Robustness	✗	✓	✓
Ease of Implementation	✓	✗	✗
Central Point of Control	✓	✗	✗
Redundancy	✗	✗	✓
Cost Effective	✗	✓	✗
Intercommunication	✗	✗	✓
Lower Complexity	✗	✓	✗
Stability	✓	✗	✓

Figure 2.5: Comparison between Control Structures

As we have already talked about decentralized, centralized and distributed, Table 2.5 will show some general features of each type.

2.6 Distributed Control and Game Theory

The relationship between distributed control problems and team decision problems has recently gained renewed attention in the engineering literature. It has been shown that a collection of controllers with access to different sets of measurements can be designed using finite-dimensional convex optimization to act optimally as a team. Game theory has been used extensively as a quantitative framework for studying communication networks and distributed control systems among its other applications in engineering and economics. Game theoretic models provide not only a basis for analysis but also for design of network protocols and decentralized control schemes [122]. The non-cooperative game theory has recently spread its use in engineering work on how to design games such that their outcome satisfies certain global objectives. Any game has three main components:

- A set of players, $N = 1,\ ...,\ n$

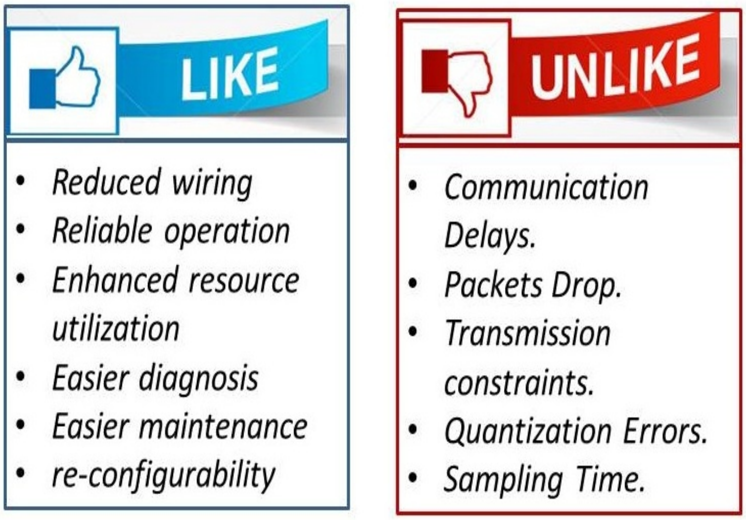
- Each player has a set of actions, A_{ij} , and the joint action space $A = A_1 A_2 ...... A_n$.
- Each player orders the outcomes according to a utility or payoff function u_i, $\forall i \in N$.

All of them are working to achieve the global objective G. When no player prefers a unilateral deviation to any of its other actions, the game is at a Nash Equilibrium [124] - [126].

As discussed in [14] by means of game theory considerations it is apparent that in iterative and independent algorithms each local controller tends to move towards a Nash equilibrium, while iterative and cooperating methods seek to achieve the Pareto optimal solution provided by an ideal centralized control structure. However, Nash equilibrium can even be unstable and far from the Pareto optimal solution, so that specific constraints have to be included in the control problem formulation to guarantee closed-loop stability. A stability constraint is included in the problem formulation, although stability can be verified only a-posteriori with an analysis of the resulting closed-loop dynamics. Then, a minmax [15] approach aimed at minimizing local cost functions under the worst-case disturbance allows one to compute parameterized distributed control laws. A team of players are to optimize a worst case scenario given limited information of natures decision for each player.

2.7 Networked Control System

The research and developments on shared data networks have a long history started by principle data networks such as Slotted and ARPANET which were specially developed around 30 to 40 years ago [123, 128]. Many industrial companies and institutes have shown interest in applying networks for remote industrial



- Reduced wiring
- Reliable operation
- Enhanced resource utilization
- Easier diagnosis
- Easier maintenance
- re-configurability

- Communication Delays.
- Packets Drop.
- Transmission constraints.
- Quantization Errors.
- Sampling Time.

Figure 2.6: Advantages and Disadvantages of NCS

control purposes and factory automation because these networks enable remote data transfers and data exchanges among users, reduce the complexity in wiring connections, minimize the costs of medias and provide ease in maintenance. As a result of extensive research and development, several network protocols for industrial control have been released, for example, Controller Area Network (CAN), Profibus, Foundation Fieldbus and Device-Net.

Most of these protocols are typically reliable and robust for real-time control purposes. Meanwhile, the technologies on general computer networks especially Ethernet have also progressed very rapidly. With the decreasing price, increasing speed, widespread usages, numerous software and applications, and well-established infrastructure, these networks became as major competitors to the industrial networks for control applications. Thus, the control applications can utilize these networks to perform remote control at much farther distances than in the past without investing on the whole infrastructure. By having the feedback control systems loops closed through a shared communication link, then it is called Networked Control System (NCS).

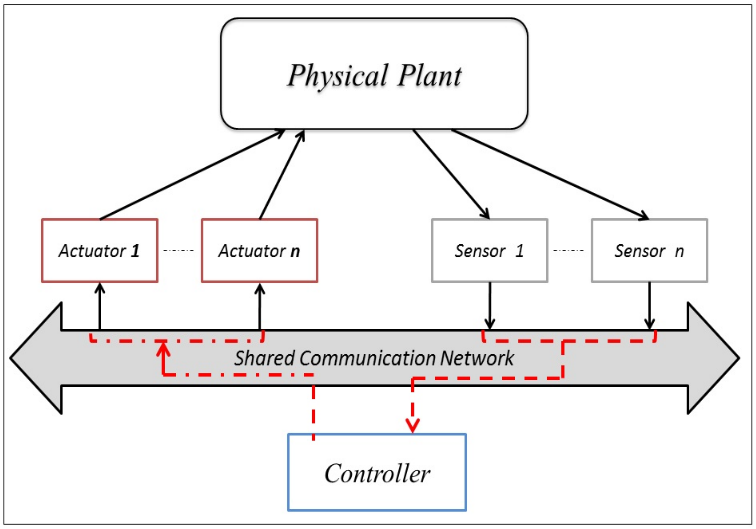


Figure 2.7: A Typical NCS Setup and Information Flows

However, the insertion of the communication network in the feedback control loop makes the analysis and design of an NCS complex. Conventional control theories with many ideal assumptions, such as synchronized control and non-delayed sensing and actuation, must be re-evaluated before they can be applied to NCSs. Specifically; the following issues need to be addressed. The first issue is the network-induced delay [74] (Sensor-to-controller delay and controller-to-actuator delay) that occurs while exchanging data among devices connected to the shared medium. This delay, either constant or time varying, can degrade the performance of control systems designed without considering the delay and can even destabilize the system [45]. Next, the network can be viewed as a web of unreliable transmission paths. Some packets not only suffer transmission delay but, even worse, can be lost during transmission. Thus, how such packet dropouts [25]-[38] affect the performance of an NCS is an issue that must be considered [8]. Another issue may occur also is when the plant outputs are transmitted using multiple network packets (so-called multiple-packet transmission), due to the bandwidth and packet size constraints of the network. Because of the arbitration of the network medium with other nodes on the network, chances are that all/part/none of the packets could arrive by the time of control calculation [5].

Depending on network protocols and scheduling methods, network-induced delays have different characteristics and can be constant, time-varying, or stochastic [6]. There are essentially three kinds of delays that will affect the system as shown in Fig. 2.8:

- Communication delay between the sensor and the controller, τ_{sc}.
- Computational delay in the controller, τ_c.
- Communication delay between the controller and the actuator, τ_{ca}.

Due to these network delay concerns, there are various methodologies which have been formulated based on several types of network behaviors and configurations to control and maintain the stability of an NCS with different ways to treat the delay problems [74]-[78], [123]-[129]. These methodologies are listed as below:

- Augmented deterministic discrete-time model methodology.
- Queuing methodology.
- Optimal stochastic control methodology.
- Perturbation methodology.
- Sampling time scheduling methodology.
- Fuzzy logic modulation methodology.
- Event-based methodology.
- End-user control adaptation methodology.
- Robust control methodology.

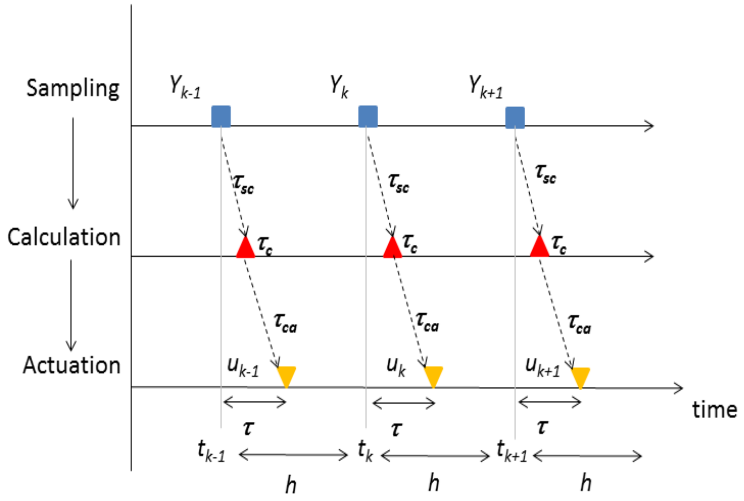


Figure 2.8: Control System with Network Induced Delays

- Time-Delay estimator methodology.
- Predictive control methodology.
- Markovian processes methodology.
- Bandwidth management methodology.

In feedback control systems, it is important that the sampled data should be transmitted within a sampling period in which the stability of control systems is guaranteed [7]. While a shorter sampling period is preferable in most control systems, for some purposes it can be lengthened up to a certain bound, maximum allowable delay bound (MADB), within which stability of the system is guaranteed in spite of the performance degradation [3]. The more information the controller can get the better control decision will be executed but this may degrade the network performance due to the high traffic generated with shorter sample period.

The NCS may have two main configurations [123, 128] or structures:

- Direct/General Structure: The structure is composed of a controller and several remote systems, each containing a physical plant, sensors and ac-

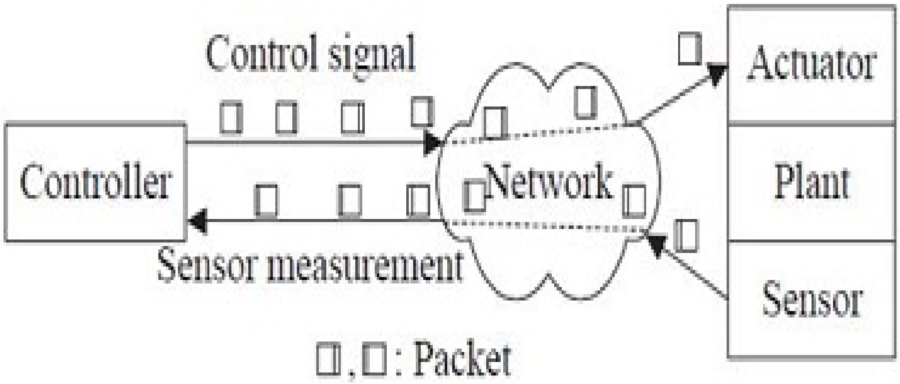


Figure 2.9: NCS in the direct structure.

tuators. The controller and the plants are physically located at different locations and are directly linked by a data network in order to perform remote closed-loop control as illustrated in Fig 2.9. In a practical implementation, multiple controllers can be implemented in a single hardware unit to manage multiple NCS loops in the direct structure. An example of this structure is the direct current (DC) motor speed control system.

- Hierarchal/Multi-Level Structure: This structure may consist of a main controller and several remote closed-loop subsystems as shown in Fig. 2.10. Each of the subsystem contains a set of sensors, a set of actuators, and a controller by itself. These system components are attached to the same control plant. In this case, a subsystem controller receives a set point from the central controller. The remote system then processes the reference signal to perform local closed-loop control and returns sensor measurement to the main controller for networked closed-loop control. Periodically, the main controller computes and sends the reference signal in a frame or packet via a network to the remote system. The networked control loop usually has a longer sampling period than the local control loop since the remote controller is supposed to satisfy the reference signal before processing the newly

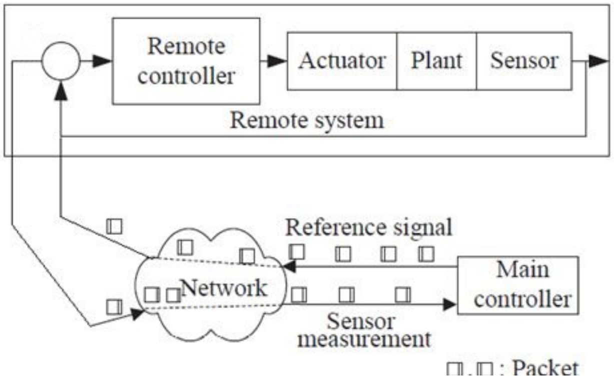


Figure 2.10: NCS in the hierarchical structure.

arrived reference signal. This structure is widely used in several applications including mobile robots and tele-operation.

Both structures have different advantages but the second structure is more modular, control loop is simpler to be reconfigured and has better interaction because data are transmitted to components directly. A controller in the first structure which can observe and process every measurement, whereas a (central) controller in the second structure may have to wait until the set point is satisfied to transfer the complete measurements, status signals, or alarm signals.

Also, when we talk about NCS, it is possible to discuss it as band-limited communication channels for control loops. The channel is digital and due to the finite word length effects only a finite number of bits can be transmitted over the channel at any transmission instant. The main issue in control (stabilization) of systems with such channels is that of quantization and we use the term quantized control systems (QCS) [155, 158] to denote systems exhibiting this feature. Another scenario if we consider the channel as a serial bus and only a subset of sensors and/or actuators can transmit their data over the channel at each transmission instant (in this case, the quantization effects are ignored). The main issue in this class of systems is time scheduling of transmissions of various signals in the system. In

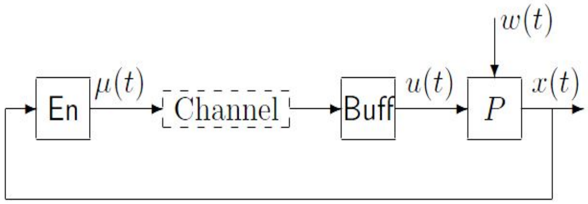


Figure 2.11: NCS subjected to Quantization

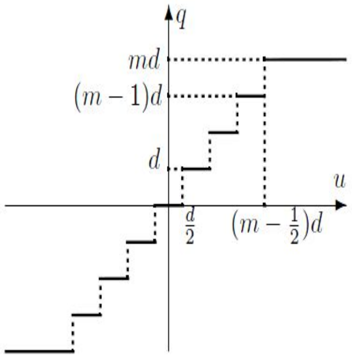


Figure 2.12: Uniform quantizer q

addition, if we are assuming that the control loop has to be implemented using a bit-rate limited channel in the plant to controller [156] communication link, then , the plant output measurements have to be quantized prior to transmission. Some ideas from the signal processing literature were borrowed and employ a feedback quantizer to encode the plant output. Using a fixed signal-to-noise ratio additive noise model for quantization errors, it is possible to show how to design the feedback quantizer to systematically reduce the impact of quantization on closed loop performance, as measured by the tracking error variance.

From Fig. 2.11 we have the plant P is a discrete-time linear time-invariant (LTI) system, The controller-encoder sends to the communication channel at time t a control packet composed of potential quantized control inputs for the current and $(N1)$ step future time instants, the quantizer q can be a static uniform quantizer (see Fig. 2.12), where the parameter d represents the step size or fineness of the quantization, and $M = 2m + 1$ is the number of the quantization levels. The buffer **Buff** decides the actuator input based on the received channel symbols. The state $b(t)$ of the buffer is updated whenever the buffer receives the packet.

Finally, NCS have been attracting significant interest in the past few years and will continue to do so for the years to come. With the advent of cheap, small, and low-power processors with communication capabilities, it becomes possible to endow sensor and actuators with processing power and the ability to communicate with remote controllers through multi-purpose networks. In view of this, we conjecture that in the near future NCSs will become the norm, replacing the current fixed-rate digital control systems that rely on dedicated connections between sensors, controllers, and actuators.

2.8 Decentralized Networked Control Systems

A system is said to be decentralized if there are multiple decision makers in the system (e.g., controllers) and these decision makers have access to different and imperfect information with regard to the system they operate in, and they need to either cooperate or compete with each other. In such control systems, one major concern is the characterization of a sufficient amount of information transfer needed for a satisfactory performance. This information transfer can be between various components of a networked control system and it will be called Decentralized Networked Control System (DecNCS). One necessity for satisfactory control performance is the ability for the controllers to track the plant states under communication constraints. One other challenge is the determination of the data rate required for the transmission of control signals, and the construction of dynamic encoding, decoding, and control policies meeting certain conditions. Another important problem is the coordination among multiple sensors or multiple con-

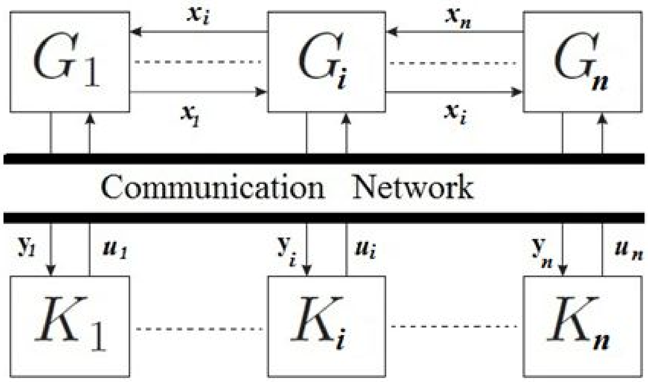


Figure 2.13: Decentralized Networked Control System

trollers/decision makers with the lowest information exchange possible which is the case in decentralized control system.

DecNCS combine the advantages of the NCS and the decentralized control systems. Such a combination enables to cut unnecessary wiring, reduces the complexity and cost of the overall system when designing and implementing control systems. DecNCS is an emerging research field for which developments are still on going to overcome several challenges raised by NCSs [41]-[43]. The control system stability in such system will require more attention to maintain the system stability [46]-[50] considering constraints like limited bandwidth or limited capacity channels, data rate constraints and multi controllers systems [44, 45].

2.9 Distributed Networked Control Systems

When the control loops are closed over a real-time or lossy communication network, then this may introduce a new term called Distributed Networked Control System (DNCS) [27]-[29] as in Fig. 2.14. In a DNCS, a given subsystem uses its state and the states of its immediate neighbors to determine its control action [30] where the information is exchanged via the communication network also. Connecting the distributed control system components via a network can effectively

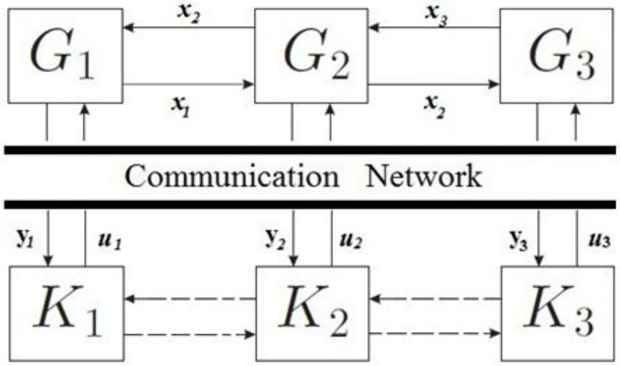


Figure 2.14: Distributed Networked Control System

reduce the complexity of the systems with nominal economical investments making the system scalable with efficient sharing of data [31]. Some parameters like induced delays, bit rate, packet size, packets drop, bandwidth and sampling time will require more focus while designing the DNCS controllers. The best examples of such system include ad-hoc wireless sensor networks and a network of mobile agents. However, the time complexity of the exact method can be exponential in the number of communication links.

The exchange of information among local controllers can be made according to different protocols:

- Information is transmitted (and received) by the local controllers only once within each sampling time (non-iterative/aperiodic algorithms).
- Information can be transmitted (and received) by the local controllers many times within the sampling time (iterative/periodic algorithms).

Going back to equation 2.5 from [27], we can see modified model considering that there is a communication between plants over a lossy communication network but it focus on packets drop only .

$$x_i(k+1) = \sum_{j=1, j\neq i}^{N} (Z_{ij} A_{ij} x_j(k)) + G_i w_i(k) \tag{2.6}$$

Where Z_{ij} is a random variable and it will be 1 at each time instant k a packet is received successfully by plant j from plant i , otherwise it is zero.

2.10 Wireless Networked Control System

Building a distributed or decentralized control system supported by a wireless network is a challenging task that requires a new design approach to both systems. Several problems, for instance, security, authentication, energy supply, signal path loss, transceiver operation mode, packet delay and dropout etc, are explored in [108] for implementation of wireless networks in industrial applications. Wireless Communication Standards like IEEE 802.11 [130], IEEE 802.15.4/ZigBee and IEEE 802.15.1/Bluetooth are used for WNCS. Most WNCS researches are based on mainly IEEE 802.11 standards and support data rates 1, 2, 11, 54 Mbps. IEEE 802.11 uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as Medium Access Control (MAC) protocol [110]. However, contention based protocols, e.g., CSMA/CA, are not appropriate for real time communication as they require handshaking among the nodes [111] and do not guarantee bounded packet delay. For the Zigbee it is used for low distance, $< 10m$, and it has two types for high data rates with QoS and low data rates with low power consumption. Bluetooth offers low cost and low power requirement with a high degree of versatility. It has been used in some industrial applications such as sensor devices for monitoring, driver hands-free calling etc. Routing protocols determines how routes are established in wireless network and can be classified into Proactive and Reactive (on demand) protocols. A proactive protocol keeps up-to-date routing table by constantly requesting update information and sharing routing tables. The disadvantage of this strategy is that it produces huge traffic in the network [112, 113]. Destination Sequenced Distance Vector routing (DSDV) is an example of proactive protocol for ad hoc networks. Reactive protocol attempts to establish a route when a node wishes to send a packet and there is no valid route in the route table. Routes are maintained until the destination becomes unreachable or the route is no longer required. The advantage is that less traffic is generated in the network. However, they have the disadvantages such as there is a delay in sending the packet and existing routes can become invalid without the node being made aware of it. Ad hoc On-demand Distance Vector (AODV) and Dynamic Source Routing (DSR) are the examples of reactive protocol.

End to end connection type Communication over wireless network can be performed using either Transmission Control Protocol (TCP) or User Datagram Protocol (UDP). TCP/IP is not suitable for it as it uses connection oriented packet transfer. On the other hand, UDP offers low overheads as it does not maintain connections and discards obsolete or lost packets. Therefore, it is preferable for networked control applications. The other issue with wireless is the security where the wireless networks inherently suffer from security problems as signals are broadcast to all receivers. Two types of security issues can be identified: Signal integrity and Authentication. For Signal integrity the main concern comes from the interference from other radio transmitters. This problem can be crucial for IEEE 802.11 and Bluetooth technology as they both use the unlicensed ISM 2.4 GHz band. However, the spread spectrum techniques implemented by the standards can mitigate the interference in most cases. Moreover, as radio signals can be received by all nearby receivers, unauthorised users can exploit the resources of WNCS. The IEEE 802.11 standard offers a WLAN authentication mechanism called Wired Equivalent Privacy (WEP) from the MAC layer. However, the security provided is not adequate. On the other hand, current Bluetooth technology specifies security in link layer and application developers have to choose the required security method. Again, Bluetooth security is not strong enough to exchange sensitive data.

2.11 Multi Agents Control System

The co-design of control, computing and communication for complex networked control systems requires a new vision on complexity and new concepts and tools that will allow the designers to analyze and simulate how timing affects control performances and to determine the optimal structure of the hybrid distributed system with computing and communication constraints. New methods based on multi-agent systems [98]-[100] could be used effectively for designing, modeling, simulating, and analyzing complex structures. Recently, the study on multi-agent systems has received more attention due to its wide potential applications, such as platooning of vehicles in the urban transportation [101, 102], the operation of the multiple robots [103], autonomous underwater vehicles [104, 105] and the formation of aircrafts in military affairs [106, 107]. Investigations for multi-agent systems begin with studying the behaviour of a large number of interacting agents with a common group objective.

2.12 Coordinated Hierarchical Control

An alternative to the distributed control schemes consists of two levels hierarchical control structures [16]-[108], like the one shown in Fig. 2.15 for the example already considered in the previous sections. In this two-level structure, an algorithm at the higher level coordinates the actions of local controllers placed at a lower level. The basic idea is to describe the overall system under control as composed by a number of subsystems linked through some interconnecting variables, i.e. the inputs of a given subsystem are the outputs or the states of another one. Then, for any subsystem an optimization problem is solved to minimize a suitable local cost function under local state, input and output constraints. If the computed local solutions satisfy the constraints imposed by the interconnecting variables, if there is coherence among the values of the interconnecting variables computed by the local controllers, the procedure is concluded. Otherwise, an iterative "price coordination" method is used: the coordinator sets the prices, which coincide with the Lagrange multipliers of the coherence constraints in the global optimization problem, by assuming as given the state, input and output variables defined by the local regulators. In turn, these optimal prices are sent to the low level local optimizers which take them as given and recomputed the optimal trajectories of the state, input and output variables over the considered prediction horizon. The iterations are stopped when the interconnecting variables satisfy the required coherence conditions. This conceptual iterative procedure must be specialized to guarantee its convergence as well as some properties of the resulting final solution.

Finally, it must be noted that similar two-level structures are widely used in the intensive stream of research in computer science and in artificial intelligence related to the so-called "autonomous agents. Basically, a number of agents must negotiate their actions through a "negotiator" until a consensus on their actions is attained, see e.g. [19]. In Fig. 2.16, a communication network is introduced in the coordinated hierarchal control system which brings all NCS issues to this type of control.

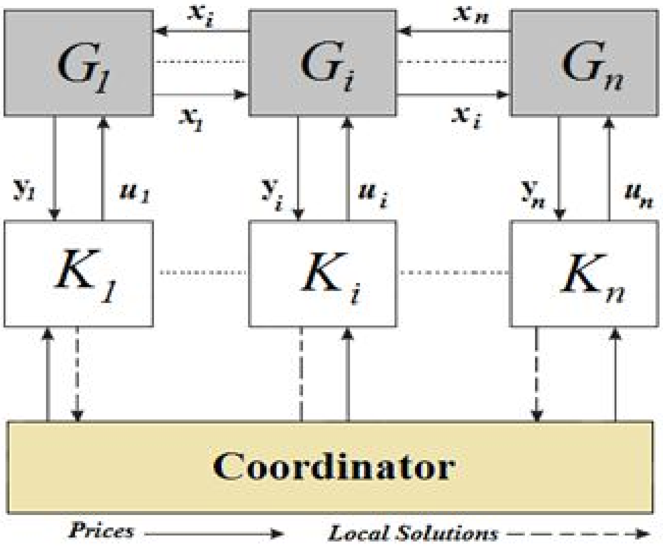


Figure 2.15: Hierarchical Control for Coordination of MIMO system.

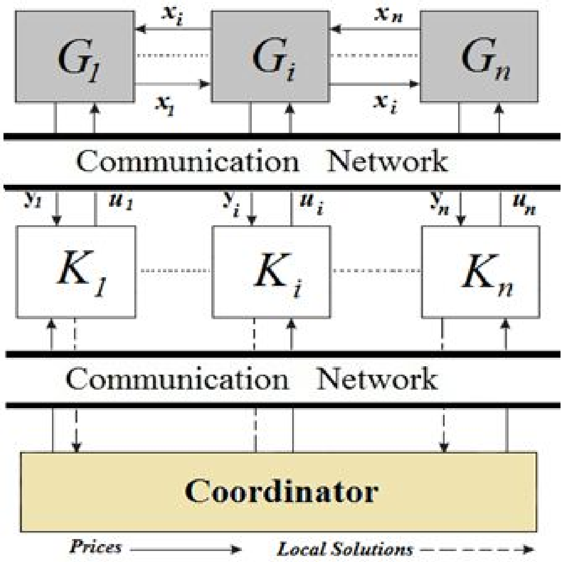


Figure 2.16: Hierarchical Networked Control for Coordination of MIMO system

2.13 Hierarchical Control of Multi Layers Systems

In hierarchical multilayer systems, the control action is performed by a number of controllers working at different time scales [9]. This can be useful at least in two cases: when the overall process under control is characterized by different dynamic behavior, i.e. by slow and fast dynamics, or in plantwide optimization when optimization and control algorithms working at a different rates compute both the optimal targets and the effective control actions to be applied. Industrial examples include a waste water treatment plant and a greenhouse control problem [20, 21, 36]. In these cases, the control can be performed at two different time scales. We can categorize the multilayer systems into the following:

2.13.1 Hierarchical control of multi time scale systems

A controller acting at lower frequencies computes both the control actions (u_{slow}) of the manipulated variables which have a long-term effect on the plant, i.e. the "slow" control variables, and the reference values of the "fast" control variables, state variables and output variables ($u_{reffast}; x_{reffast}; y_{reffast}$), respectively. A second controller takes these reference values as inputs and computes the "fast" control variables u_{fast} solving a tracking problem at a higher rate. A conceptual scheme of this architecture for a two-layers structure is shown in Fig. 2.17.

2.13.2 Control of Systems with Hierarchical Structure

Many industrial, economical or sociological systems can be described by a hierarchical structure where the highest layer of the hierarchy corresponds to a dynamical system with slow dynamics. This system can be controlled by looking at its behavior over a long time scale, and its computed control inputs must

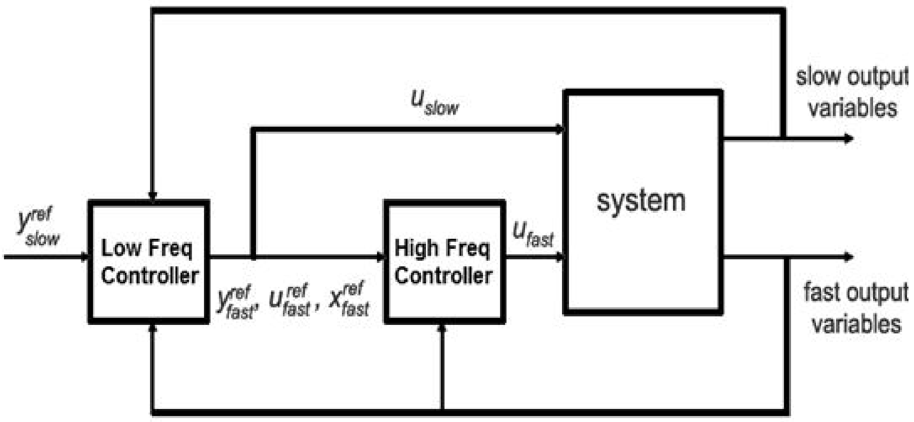


Figure 2.17: Control of System with Slow and Fast Dynamics

be effectively provided by subsystems placed at lower layers of the hierarchy and characterized by faster dynamics. In turn, these subsystems must be controlled at a higher rate and can be placed at an intermediate layer of the hierarchy. An example of a three layer structure is shown in Fig. 2.18. As a matter of fact, in these structures the regulator at a higher layer computes its desired control inputs, which are the reference signals of the immediately lower layer [160]. Moreover, the controllers of the subsystems at the lower layer must guarantee the solution of the corresponding tracking problems with an adequate level of accuracy, so that the mismatch between what is required by the higher level and what is provided by the lower one does not destroy some fundamental properties, such as stability and performance. From a control engineering point of view, this multilayered hierarchical structure corresponds to a classical cascade feedback control system as in Fig. 2.19.

2.13.3 Hierarchical Control for Plant-wide Optimization

In the process industry it is common to design the overall control system according to the hierarchical structure shown in Fig.2.20. At the higher layer, real

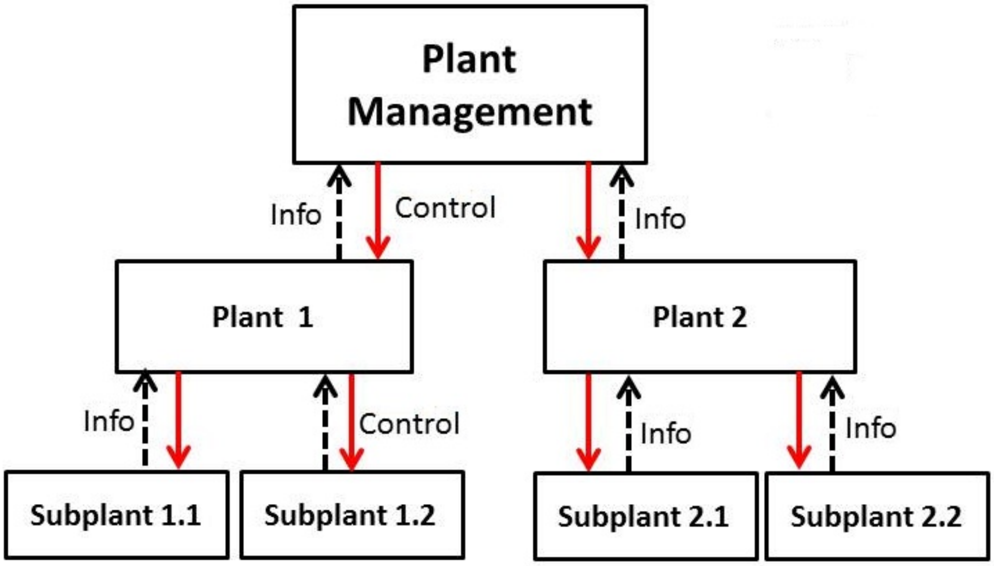


Figure 2.18: Hierarchical Structure of a Three Layer System

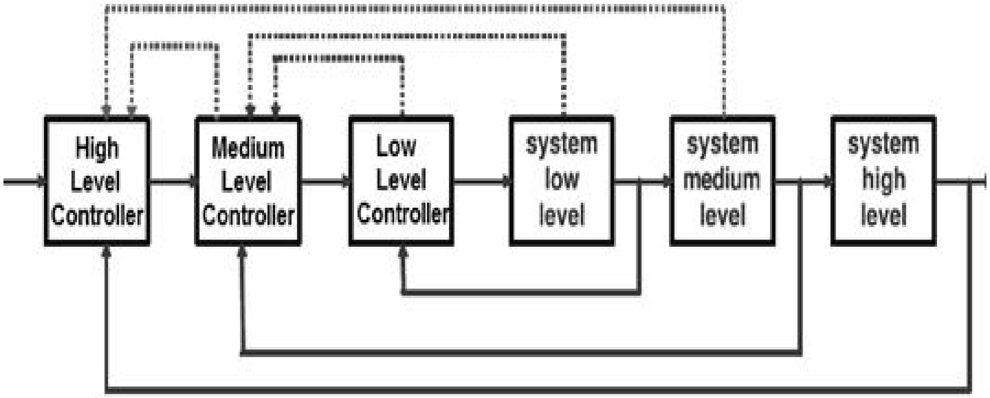


Figure 2.19: Three Layers Cascade Control Structure

time optimization (RTO), [160] which plays a fundamental role in process control, is performed to compute the optimal operating conditions with respect to a performance index representing an economic criterion. At this stage a detailed, although static, physical nonlinear model of the system is used. At the lower layer a simpler linear dynamic model of the same system, often derived by means of identification experiments, is used to design a controller guaranteeing that the target values transmitted from the higher layer are attained. Also in this case, the lower level can transmit bottom-up information on constraints and performance. Moreover, the controller design shall take care of constraints arising from closing the control loop over a shared communication network.

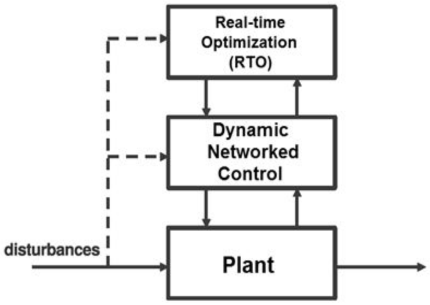


Figure 2.20: Hierarchical Structure for Plant-wide Control and Optimization

2.13.4 Hierarchical Control System for Dynamic Resource Management

The hierarchical control system uses a set of utility functions to evaluate the performance of strings and missions in the system against current resource allocations [55]. The control system also uses the utility estimation function to estimate the desirability of various control actions with respect to the future performance and utility of the system. The control system chooses control actions that would result in a higher level of estimated utility [16]. If the system has enough unused system resources, the system could allocate resources to previously un-deployed missions or application strings to boost its overall utility and performance. Conversely, if resource contention were occurring due to an over deployment of missions (possibly due to resource failure among other possible causes), then the performance and utility of the deployed missions would drop. A drop in the measured utility indicates to the controllers that the allocation of resources should be adjusted in an attempt to relieve the resource contention and raise the measured utility. The control system uses a hierarchical control [22] philosophy that is fundamentally bottom-up. The low level controllers are generally fast and responsive, while the high level controllers have the ability to take more aggressive control actions. Because higher level control actions are more invasive, the higher level controllers take more time to better estimate which of their control actions will maximize their local utility. Because local controllers in this design attempt to greedily maintain their local utility, the bottom-up control philosophy limits local, fast utility gains that are potentially detrimental to the overall system utility.

String controllers perform fast low-level tuning of quality and throughput in order to maintain their local string utility. If a string controller is unable to maintain its local utility, its mission controller performs limited resource re-deployments to benefit local strings. If a mission controller is unable to maintain its local utility, the mission controller sends a request to the system controller to re-initialize system resources. The system controller has the ability to request that the Infrastructure Allocator perform a full re-initialization of system resources if the controllers are unable to take any action that would sufficiently raise the measured system utility. The controllers interact with each other through direct communications, but the controllers receive information about system resources or performance through resource status.

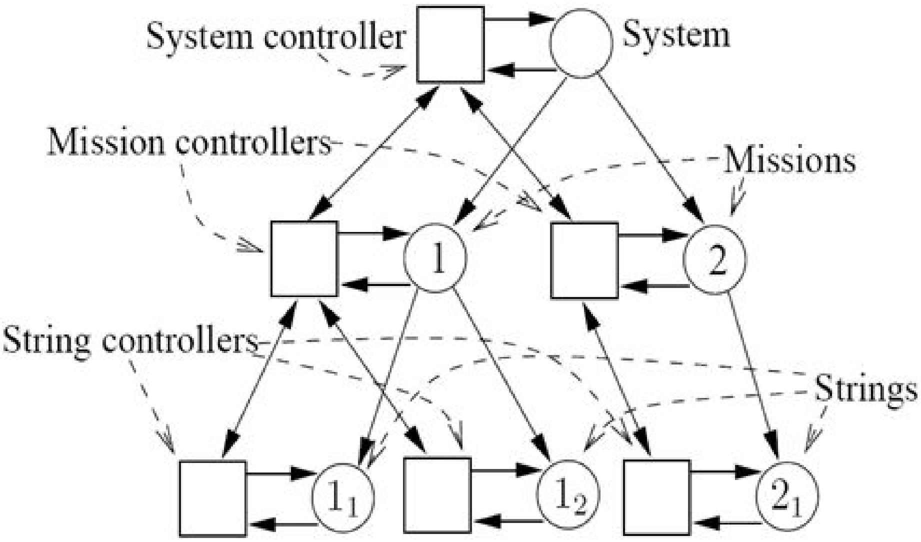


Figure 2.21: The Control Hierarchy

CHAPTER 3

OBSERVER-BASED DECENTRALIZED NETWORKED CONTROL

3.1 Overview

In this chapter, we investigate a decentralized control approach to dynamical systems where the control loops are spread over a network. In such a decentralized networked control systems (DecNCS), the subsystems are communicating to each other over a shared communication network. The properties of the network such as delay, packet dropout, varying sample interval and induced errors are considered in the design of the control system. These properties add restrictions and difficulties in the control loop that are not present in traditional control loops. Also, the general structure of a decentralized networked control system is described, the main characteristics, major problems, network communication parameters and the techniques of handling lost control data. Simulation applied on standard system and numerical verification also presented.

3.2 Introduction

A decentralized networked control system (DecNCS) is a class of decentralized control systems in which the different components (units) are connected over shared communication channels or a data network. It is envisioned that there is a data link between the sensors (which collect information), controllers (which make decisions), and actuators (which apply the controller commands) [42, 43]. In general, a system is said to be decentralized if there are multiple decision makers in the system (for example, sensors, controllers, encoders)and these decision makers have access to different and imperfect information with regard to the system they operate in, and they need to either cooperate or compete with each other. Such systems are becoming ubiquitous, with applications ranging from automobile and inter-vehicle communications design, control of surveillance and rescue robot teams for access to hazardous environments, space exploration and aircraft design, among many other fields of applications [44]-[46]. In such control applications, one major concern is the characterization of a sufficient amount of information transfer between/ various components of DecNCS (see Fig. 3.1) needed for a satisfactory performance. Several necessities for satisfactory control performance are the ability for the controllers to track the plant state under communication constraints, the determination of the data rate required for the transmission of control signals, and the construction of dynamic encoding, decoding, and control policies meeting some criteria [47]. Another important problem is the coordination among multiple sensors or multiple controllers/decision makers with the lowest information exchange possible. Even in cases when communication resources are not scarce, a strong understanding of the fundamentals can be useful in the system architecture, and finally, such an insight can help reduce the computation requirements and complexity.

The insertion of a communication network can substantially improve the flex-

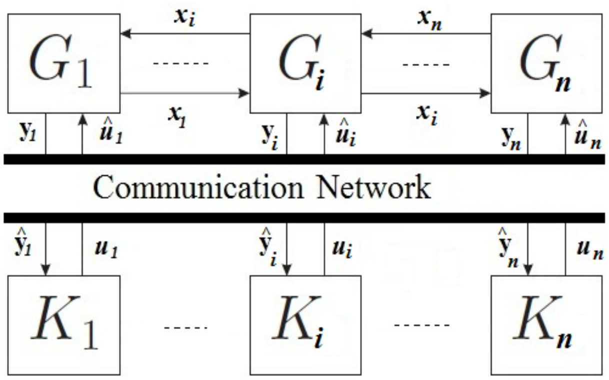


Figure 3.1: A decentralized networked control system

ibility and fault-tolerance of an industrial control system, in addition to reducing the installation, material/labor cost, reconfiguration, maintenance time and costs [1, 5]. Currently, process control systems utilize dedicated,wired control networks to achieve key closed-loop properties like stability, set point tracking and robustness to disturbances. Robustness and reliability are major concerns because the interference in the process control field and the consequence of a failure can be severe. Interference caused by environmental events and other signals impacts timely data transmits which directly challenges the objective of real-time process control [25, 132]. Timing scheme uses clock-driven sensing and actuation with event-driven control and the control loop can adapt to varying network conditions [86]. Moreover, the network-induced delay considered in [13] is composed of the sensor-to-controller delay and the controller-to-actuator delay as well as the computation delay and can be slowly or quickly time-varying. Then, a controller design method is proposed based on a delay-dependent approach. The effect of unreliable channels have on overall system performance assuming that there are no quantization errors been shown in [44]. The optimal H_2 design of semi-decentralized controllers [26, 133] is considered for a special class of spatially distributed systems. This class includes spatially invariant and distributed discrete-time systems with an inherent temporal delay in the interaction of neighboring sites [134]. Another approach introduces robust stabilization of discrete-time delay systems under non-linear perturbations and this is transformed to a constrained convex optimization. Sufficient conditions on the existence of state feedback controllers are established in terms of Linear Matrix Inequality (LMI), which guarantee stability of the closed-loop system and at the same time, maximize the non-linearity bound [121].

For a group of Lagrangian vehicle systems with directed communication graph topology, the cooperative tracking control problem is investigated in [136] where all the vehicles can have different dynamics. A design method for a distributed adaptive protocol is given which guarantees that all the networked systems synchronize to the motion of a target system. A methodology to control multi-agent systems is provided in [137] where the stability of evolving agent populations is investigated through simulation. In [138], the problem of stability analysis is investigated for switched neural networks with time-varying delay by taking advantage of the average dwell time method. Using linear matrix inequality (LMI) approach, two sufficient conditions are developed to ensure the global exponential stability of the considered neural networks. A decentralized H_∞ fuzzy filter design for non-linear interconnected systems with multiple time delays via T-S fuzzy models is introduced in [139]. The asymptotic stability and a prescribed H_∞ performance index are guaranteed for the overall filtering error system. A guaranteed cost networked control (GCNC) method for T-S (Takagi Sugeno) fuzzy systems with time delays is developed in [140], where the state feedback controller is designed via the networked control system (NCS) theory. The problem of guaranteed cost control for TS fuzzy dynamic systems with interval parameter uncertainties is further investigated in [141] based on the instrumental idea of delay dividing.

In this work, we consider several network side effects because the controllers are communicating to the plants over a shared communication network. The basic features of the controller will be used are discrete, decentralized and output observer based type. The controller will do a control **over network** not through network and design shall consider the following:

1. Uncertain time delays due to communication, processing and queuing.
2. Transmission constraints where not all outputs and inputs can be transmitted at the same time.
3. Quantization Error.
4. Fixed and Varying Sampling Intervals.
5. Sampling interval selection.
6. Unreliable transmission and Packets dropout.
7. Network induced Errors.
8. Interconnected Communication.

3.3 State-Feedback Control

State feedback is a time-domain based approach to controller design using state-space plant models. It was very popular among control researchers in the sixties and the seventies, which resulted in an impressive body of knowledge including optimal control. Perhaps one of the biggest benefits of state feedback over classical loopshaping was that it could directly handle MIMO plants.

One of the original objectives of state feedback was pole placement. We know that the eigenvalues of the A matrix for a state-space system (A, B, C, D) correspond to the poles of its transfer function

$$G(s) = C(sI - A)^{-1}B + D \tag{3.1}$$

So the idea was to use the state vector as the input to a constant controller matrix K is order to change the location of the eigenvalues of A to other more desirable locations in the complex plane. A more remarkable result is the Linear Quadratic Regulator (LQR) which is an optimal state feedback controller with respect to a particular objective function. Of course the state variables can't always be measured in practice. This led to the development of state observers whose purpose was to compute an estimate of the state vector which was then used in a regular state feedback configuration. The Kalman filter, bearing the name of its inventor, is an optimal state observer which was developed in the setting of stochastic systems. It can be used in a deterministic setting as well.

Finally, the combination of LQR state feedback and the Kalman filter forms what is called the Linear Quadratic Gaussian (LQG) controller, which represented the culmination of two decades of research on the state-space approach to control design. A problem remained with LQG controllers though, and this was noticed in the early eighties: they are not robust. This is why their acceptance in industry had been somewhat slow.

let us consider the state-space system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{3.2}$$

A typical block diagram for a state feedback regulator on this plant model is shown in Fig. 3.2.

The constant real matrix K multiplies the state vector to generate the control signal $u(t)$ in the state feedback law $u = -Kx$. One can readily see that the setup in Fig. 3.2, violates the principle that only measured signals collected in $y(t)$ are available for feedback. Therefore, state feedback can be directly applied

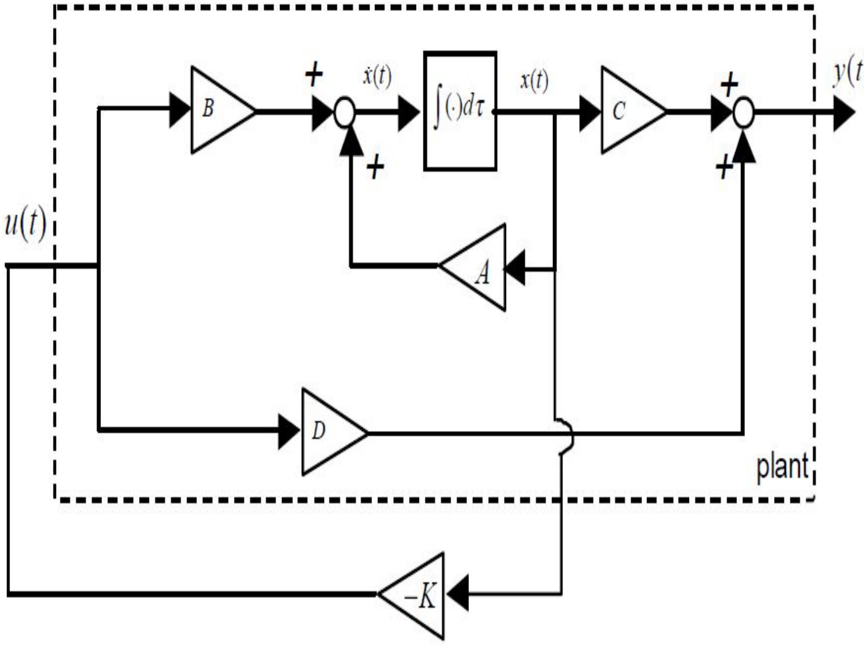


Figure 3.2: State feedback diagram

only if the C matrix is square and invertible.

Another issue need to be mentioned here which is the Controllability that addresses the issue of whether the set of actuators can control the state of the plant.

Definition 3.1 *An LTI state-space system is controllable if, given any constant target state* x_1 *and any time* T *, there exists an input signal* $u(t)$*,* $0 \leq t \leq T$ *taking the initial state* $x(0) = 0$ *to* $x(T) = x_1$ *. If a process is uncontrollable, then there isn't much one can do, apart from redesigning the process itself or adding actuators to obtain controllability.*

Controllability of the state-space system (A, B, C, D) can be tested using the following result: (A, B, C, D) is controllable iff the controllability matrix $C :=$ $[B\ AB ......\ A^{n-1}B]$ has full rank n .

Another issue is the Observability which addresses whether the set of sensors can "observe" the state of the plant.

Definition 3.2 *An LTI state-space system* (A, B, C, D) *is observable (or, in short, the pair* (A, C) *is observable) if, given any initial state* $x(0) = x_0$ *, the initial state can be uniquely reconstructed from knowledge of the input* $u(t)$*,* $0 \leq t \leq T$ *and the output* $y(t)$*,* $0 \leq t \leq T$ *for any time* $T > 0$ *.*

Again, if a process is unobservable through the sensors, then there may be state trajectories that don't appear at the measured output. Hence the process can't be properly monitored or controlled. One would need to redesign the process or add sensors in judicious locations to obtain observability.

Observability of the state-space system (A, B, C, D) can be tested using the following result:

(A, B, C, D) is observable iff the observability matrix $[\ C\ CA\ ....\ CA^{n-1}\ ]'$ has full rank n .

3.4 Observer-Based Control

The objective of state estimation is, as the name implies, to provide an estimate $\hat{x}(t)$ of the state vector $x(t)$ from measurement of the output $y(t)$. An estimate of the state of a system finds different applications in industry such as plant monitoring (smart sensors), fault detection, navigation and obviously state feedback. The term observer is used for state estimators as they "observe" the state through the output of the system. We will deal with deterministic (as opposed to stochastic) observers. Even the Kalman filter, which is an optimal stochastic observer, can be derived in a deterministic framework, making things much simpler.

The state observer produces an estimate of the entire state vector from measurement of the output and input signals. Consider the state-space system representing the plant whose state is not completely measured, and for which both the state and the output are corrupted by the deterministic noise signals w and v

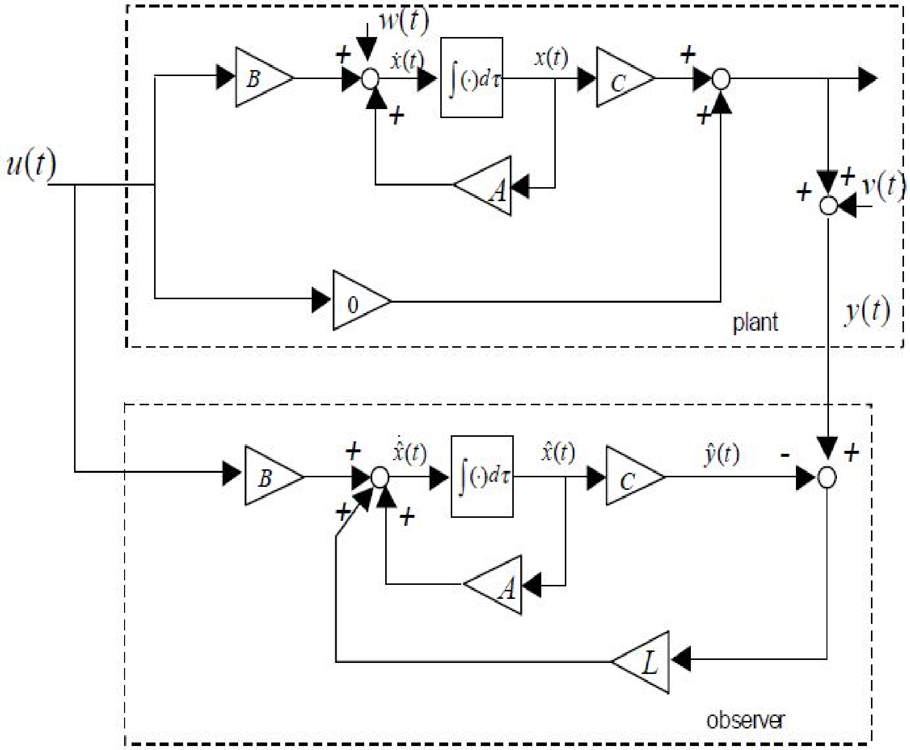


Figure 3.3: State Observer diagram

, respectively:

$$\begin{aligned} \dot{x}(t) &= Ax + Bu + w \\ y &= Cx + v \end{aligned} \tag{3.3}$$

Assume that we know the state-space matrices of the plant with perfect accuracy. The state-space system describing the dynamics of the observer is as follows:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{3.4}$$

which can be rewritten as the following;

$$\begin{aligned} \dot{\hat{x}} &= (A - LC)\hat{x} + Ly + Bu \\ \hat{y} &= C\hat{x} \end{aligned} \tag{3.5}$$

The goal is to design the observer gain L such that the state estimate will track the state. This can be expressed in terms of the state-space system governing the evolution of the error:

$$e(t) := x(t) - \hat{x}(t) \tag{3.6}$$

A bit of algebra shows that this system is given by:

$$\dot{e}(t) := (A - LC)e + w + Lv \tag{3.7}$$

Therefore, it suffices to find a matrix L such that all the eigenvalues of $(A-LC)$ (the poles of the observer) are in the open left half-plane to ensure that the error will tend to zero when the noises are zero. Note that this is true even for quickly varying inputs since (3.7) doesn't depend on the input signal. A fast observer is obtained with a "large" matrix gain L . For systems with a single output, the technique of pole placement can be used to design L (a column vector) with the difference that the state-space system $(A, B, C, 0)$ should be expressed in an observable canonical form first.

3.5 Problem Definition

We consider a class of linear continuous-time systems represented by:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3.8}$$

where $x(t) \in \Re^n$ is the state vector; $u(t) \in \Re^m$ is the control input, and $y(t) \in \Re^q$ is the output vector, which represents the real measured output taken directly from the plant G. This is an ideal presentation of system model without the presence of the network which is our focus in this study. The same model in (3.2) will have some changes in the presentation due to the network effects. For example, the actual control input will be $\hat{u}$ that control the plant after considering the network effects on it or we can call it the network version of original control signal u. Similarly, the output measurement that sent from the plant back to the controller over the network will be $\hat{y} = C\hat{x}$ which is the network version of y.

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B\hat{u}(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3.9}$$

All decentralized controllers $K_i,\ i = 1,\ ...,\ n$, are communicating with sensors and actuators over a shared network. Each plant G_i is controlled by discrete time observer based controller. The plant model which is described in (3.3) will be discretized with a zero order hold to:

$$\begin{aligned} x_{k+1} &= A_d x_k + B_d u_k \\ y_k &= C x_k \\ A_d &= e^{Ah}, \quad B_d = [\int_0^h e^{As}\, ds\,]B \end{aligned} \tag{3.10}$$

where $x_k = x(t_k) = x(kh),\; y_k = y(t_k) = y(kh)$ and t_k are discrete points of time and k is an integer time index. The distance in time between each point of time is the time-step, denoted here by h, which is the sampling interval $h = t_k - t_{k-1}$. The discrete time control input $\hat{u}_k$ is available at the plant at time $t = t_k$.

Basically what is modeled up to now is for one system and since we have a number of systems connected in decentralized structure then we need to consider the interconnection term as the following with the superscript i used to indicate subsystem i:

$$\begin{aligned} x^i_{k+1} &= A^i_d x^i_k + B^i_d \hat{u}^i_k + \sum_{j=1, j\neq i}^{N} A^{i,j}_d \hat{x}^j_k \\ y^i_k &= C^i_d x^i_k + \sum_{j=1, j\neq i}^{N} C^{i,j}_d \hat{x}^j_k \\ \hat{u}^i_k &= K^i \hat{x}^i_k + \sum_{j=1, j\neq i}^{N} K^{ij} \hat{x}^j_k \end{aligned} \tag{3.11}$$

The interconnection term $A^{i,j}_d \hat{x}^j_k,\; j \neq i$ describes how the dynamics of the i^{th} unit are influenced by the j^{th} unit in the plant. Note from the summation notation that each processing unit can in general be connected to all other units in the plant. Also, the system is decentralized which means that the interactions between subsystems shall be as minimum as possible to make the decentralized design successful. The term $K^i \hat{x}^i_k$ in the control law equation represents the local feedback component responsible for stabilizing the i^{th} subsystem in the absence of interconnections, and $\sum_{j=1,j\neq i}^{N} K^{ij}\hat{x}_k^j$ is a "feed forward" component that compensates for the effect of the j^{th} neighboring subsystem on the dynamics of the i^{th} unit. Note in this regard that the implementation of the control law requires the availability of state measurements from both the local subsystem that is being controlled as well as the other connected units. It is significant to observe that a choice of $K^{ij} = 0$ reduces the control strategy to a fully decentralized one where only measurements of the process variables of the i^{th} unit are collected and processed with no signal transfer taking place across the network from other systems.

All communication between sensors, actuators and controllers are going over a shared communication network. The sensors and controller nodes are clock-driven while actuator is event-driven and it can be also a clock-driven. This means that the controller will not compute new control command until it receives the sensors' measurements and the actuator continues using the old command until new one has arrived. By event-driven we mean that the node starts its activity when an event occurs, for instance, when it receives information from another node over the data network. Clock-driven means that the node starts its activity at a pre-specified time, for instance, the node can run periodically. In the following subsections we will highlight the impacts of having the communication between the system components over a shared network and try to explain each effect separately with details.

3.6 Network Effects

3.6.1 Induced Delays

Communication over shared network causes time delays in various sections of Dec-NCS. These time delays cannot be neglected, especially when the time constant of the controlled plant is short and the order of the plant model is high because they may affect network QoS and degrade control performance. The delays can be classified into different types using different classification criteria. It can be categorized from the direction of data transfers as the sensor-to-controller delay, computation delay in the controller and the controller-to-actuator delay. Also, it can be separated as device delay and network delay where the device delay is divided into several subtypes of delays. There are several factors that may affect the network time delays like the network load, network schedule, network bandwidth, size of the messages and message priority. Back to delays classifications, in the first classification, there are essentially three kinds of delays that will affect the system:

- Communication delay between the sensor and the controller, τ_{sc}.
- Computational delay in the controller, τ_c.
- Communication delay between the controller and the actuator, τ_{ca}.

For a discrete-time system representation we can use the following indications, τ_{sc}^k, τ_c^k, τ_{ca}^k to show the delay at certain time instant k. The computational delay of the controller is very small and it can be neglected or added it as a part of τ_{ca} delay. As a result, the round trip time for packet in control loop can be shown as sum of τ_{sc} and τ_{ca}, which we have used in our work, that is:

$$\tau_{rt}^k = \tau_{sc}^k + \tau_{ca}^k \tag{3.12}$$

Another way is to calculate τ_{ca} and τ_{sc}, as shown in 3.13.

$$\tau_{sc} = \tau_{cs} - \tau_{se}, \quad \tau_{ca} = \tau_{as} - \tau_{ce} \tag{3.13}$$

where

- τ_{se} : time instant when remote sensor encapsulates the measurement into sent packet,
- τ_{cs} : time instant when controller starts processing the delivered measurement packet,
- τ_{ce} : time instant when main controller encapsulates the control signal into sent packet,
- τ_{as} : time instant when remote system starts processing the control signal.

Moreover, the delays τ_{sc} and τ_{ca} are composed of at least the following parts:

- *Waiting time delay* τ_w : is the time of which a source (the main controller or the remote system) has to wait for queuing and network availability before actually sending a frame or a packet out. Also it can be called queuing delay τ_q.
- *Frame time delay* τ_F : is the time during the moment that the source is placing a frame or a packet on the network.
- Propagation delay τ_P : is the delay for a frame or a packet traveling through a physical media. It depends on the speed of signal transmission and the distance between the source and destination.

These delay parts are fundamental delays that occur on a communication network. When the control or sensory data travel across networks, there can be additional delays such as the queuing delay at a switch or a router, and the propagation delay between network hops. The delays τ_{ca} and τ_{sc} also depend on other factors such as maximal bandwidths from protocol specifications, and frame or packet sizes.

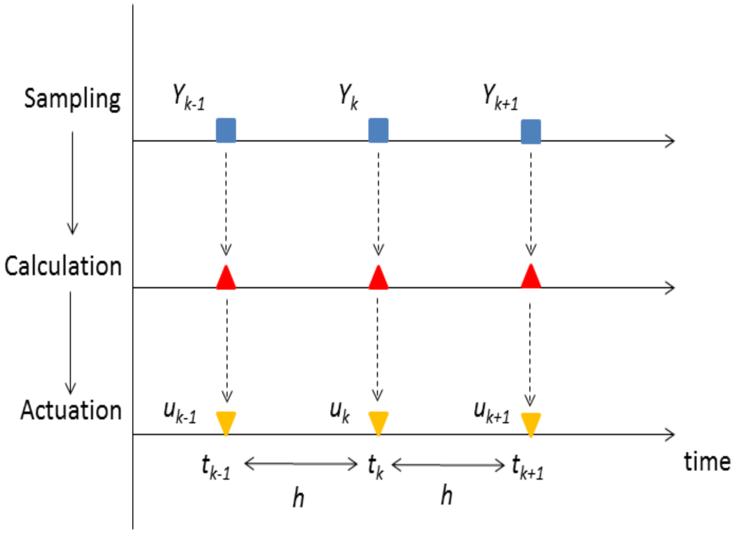


Figure 3.4: Control system with instantaneous input-to-output latency

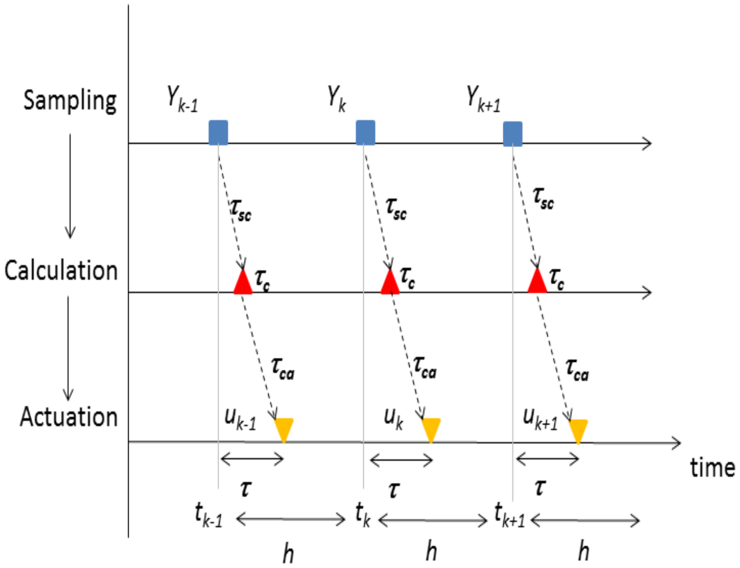


Figure 3.5: Network induced delays that make input-to-output latency > 0

Figs 3.4 and 3.5 show systems with short or constant time delays, which means that the delay is less than one sampling period and this, makes (3.10) in the following form:

$$\begin{aligned}
x_{k+1} &= A_d x_k + B_{d0}(\tau_{rt}^k)\hat{u}_k + B_{d1}(\tau_{rt}^k)\hat{u}_{k-1} \\
y_k &= C x_k \\
A_d &= e^{Ah}, \\
B_{d0} &= \int_0^{h-\tau_{rt}^k} e^{As}\, ds\, B, \\
B_{d1} &= \int_{h-\tau_{rt}^k}^{h} e^{As}\, ds\, B
\end{aligned} \tag{3.14}$$

The control samples u_k and u_{k-1} were applied during the $[k-1, k]$ period (see Fig. 3.5) and it is shown as a network version since it is being sent over a network. Considering the N interconnected term, then (3.14) will be:

$$\begin{aligned}
x_{k+1}^i &= A_i x_k^i + \Phi_i \hat{u}_k^i + \Theta_i \hat{u}_{k-1}^i + \sum_{j=1, j\neq i}^{N} A_{i,j}\hat{x}_k^j \\
\hat{y}_k^i &= C_i x_k^i + \sum_{j=1, j\neq i}^{N} C_{i,j}\hat{x}_k^j \\
\hat{u}_k^i &= K^i x_k^i + \sum_{j=1, j\neq i}^{N} K^{ij}\hat{x}_k^j
\end{aligned} \tag{3.15}$$

where $\Phi_i = B_{d0}(\tau_{rt}^k),\ \Theta_i = B_{d1}(\tau_{rt}^k),\ A_i = A_d^i$ are used for simplicity in exposition. It must be observed that we have not considered the delay explicitly in the interconnection term due to the fact that if it was not occurring within the same sampling period, it will be eventually ignored. In general, if the delay is less than one sampling interval h for the system i, we can estimate the maximum delay as h. However, when the delays are longer than one sampling interval h, in this case the delayed packets will be drop.

3.6.2 Packets Dropout

From (3.11), if the delay $\tau_{rt}^{k} > h$ for a packet, then it will be lost. This means that it is not arrived before the end of the sample period. Let us introduce two parameters to be used for the packets drop formulation, namely α_k and β_k, where $\alpha_k,\ \beta_k \in \{0, 1\}$. At time instant k the following could happen:

- Sensor packet is lost, which implies that $\beta_k = 0$,
- Control command packet is lost, which implies that $\alpha_k = 0$

The assumption used here is that the actuator uses the previous control command $\hat{u}_{k-1}$ if the current control command u_k has not reached. In normal situation, it will use the most recent control input that remains active until new one arrives. If an actuation packet lost and the controller is passive [142], this implies that it will provide actuation when a packet is received from sensor. The same took place for the measurement data y_k if not received then the previous data will be used $\hat{y}_{k-1}$. These can be represented by the following:

$$\begin{aligned} y_k &= \beta_k \hat{y}_k + (1-\beta_k) y_{k-1} \\ \hat{u}_k &= \alpha_k u_k + (1-\alpha_k)\hat{u}_{k-1} \end{aligned} \tag{3.16}$$

When the control packet is lost, the DecNCS model described in (3.15) will be

$$\begin{aligned} x_{k+1}^{i} &= A_i x_k^i + (B_{d,prev})_i \hat{u}_{k-1}^{i} + \sum_{j=1, j\neq i}^{N} A_{i,j}\hat{x}_k^j \qquad (3.17) \\ (B_{d,prev})_i &= [\int_0^h e^{As} ds]B \end{aligned}$$

Furthermore, the number of subsequent packet dropouts is upper bounded by ε and guarantees that from the sequence of previous control inputs

$\{u_{k-\varepsilon},\ u_{k-\varepsilon+1},\ ...\}$ at least one is implemented. This means that

$$\sum_{i=k-\varepsilon}^{k} \alpha_k \ \leq \ \varepsilon \tag{3.18}$$

3.6.3 Induced Errors

The network induced error can simply be shown as discrepancies between the current and most recently transmitted input/output values of nodes' signals and it can be used as shown in [147], to design dynamic output feedback and communication protocol without the need for any knowledge about the controller and plant states. Also, the network induced error can be used for transmission scheduling where the node with highest error will have the highest chance to obtain the network access for transmission.

$$e_k^u \ = \ \hat{u}_{k-1} \ - \ u_k, \quad e_k^y \ = \ \hat{y}_{k-1} \ - \ y_k \tag{3.19}$$

We can define threshold levels $\gamma_i^u,\ \gamma_i^y$ for the induced error based on (3.19) where $e_k^u \ < \ \gamma_i^u$ and $e_k^y \ < \ \gamma_i^y$ for each subsystem i.

3.6.4 Quantization Errors

In modeling communication channels capacity and buffers for control loops, the channel can be digital and due to the finite word length effects only a finite number of bits can be transmitted over the channel at any transmission instant. The main issue in control systems with such channels is that of quantization because the use of quantizer will add a quantization error and this error will vary based on different parameters. The resulted quantization error is often modeled as uniform and white noise [134]. Also, when we have to convert A/D or D/A during the communication between sensors/controller and controller/actuator we need to use

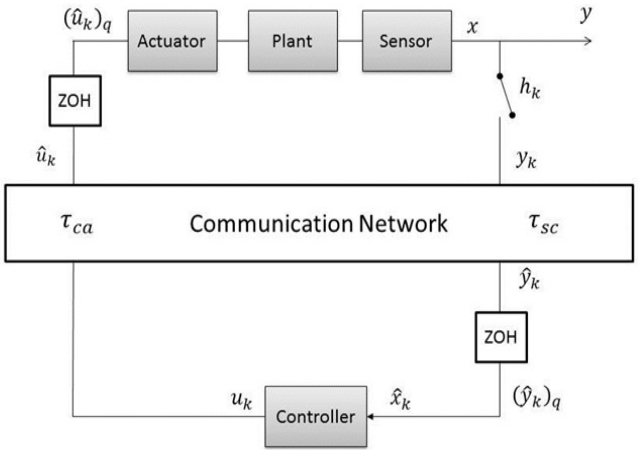


Figure 3.6: Control system with delays, quantizer and varying sample interval

the quantizer [135]. Designing the quantizer can impact the system performance in different manner. If the quantization regions are defined so that they do not change with time then this is a Static Quantizer [157]. This type is simple to implement in both hardware and software and not computationally expensive. It has two types, namely uniform quantizers and logarithmic quantizers. The dynamic quantization has a varying quantized region and quantization error at each transmission time which will make the quantizer more complex and needs to compute new quantization regions and detect the plant state presence within this region each time.

From Fig. 3.6, we have the networked and quantized version of actual input u which is $\hat{u}_q$, the networked version of the system state $\hat{x}$ and $\hat{y}$, where the zero-order hold (ZOH) function is applied to transform the discrete-time control input to a continuous-time control input being the actual actuation signal of the plant.

$$u = u_k = K\,\hat{x}_k, \;\; (\hat{u}_k)_q = f(\hat{u}_k) \tag{3.20}$$

The quantization error can be defined as the difference between u_q and u:

$$\begin{aligned} (e_k)_q &= (\hat{u}_k)_q \; - \; u_k \; = \; f(\hat{u}_k) - K \, \hat{x}_k \\ &= \Delta_k \, f(\hat{u}_k) \end{aligned} \tag{3.21}$$

where Δ_k is a quantized error [159] parameter that is bounded by a certain threshold value, and comparing to other NCS issues, the quantization is too small to be considered, so it can be neglected.

3.6.5 Variable Sampling Interval

Due to the nature of the network, the actual sampling times are not necessarily equidistant in time. In (3.3) we have used a constant sampling interval h but this actually will vary for each instant k to be h_k , see Fig. 3.4, and hence (3.3) becomes

$$\begin{aligned} x^i_{k+1} &= A^i_p x^i_k + B^i_p \hat{u}^i_k + \sum_{j=1, j\neq i}^{N} A_{i,j} x^j_k \\ y^i_k &= C^i_p x^i_k + \sum_{j=1, j\neq i}^{N} C^{i,j}_p x^j_k \\ A^i_p &= e^{A h_k}, \quad B^i_p = [\int_0^{h_k} e^{As} ds] B \end{aligned} \tag{3.22}$$

where A^i_p and B^i_p are basically the discretized version of A_d and B_d that were mentioned earlier for constant sampling interval. Here however, they come with varying sample interval. The state measurements are sampled at the sampling times t_k given by:

$$t_k = \sum_{i=0}^{k-1} h_i, \quad \forall \, k \, > 0 \tag{3.23}$$

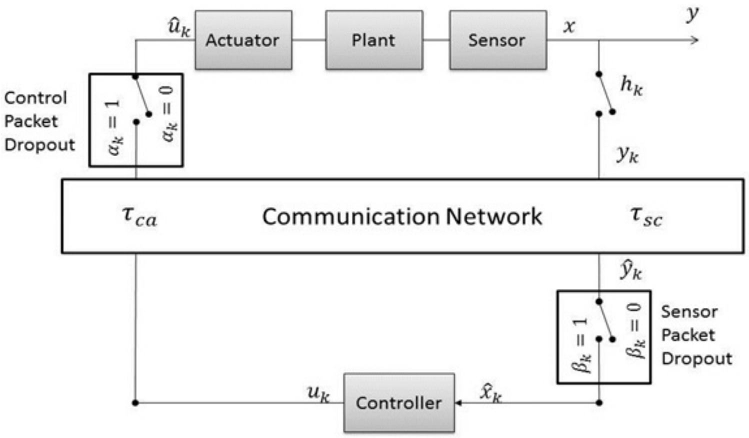


Figure 3.7: DecNCS with packet dropout and varying sample interval

which are non-equidistantly spaced in time due to the time varying sampling intervals $h_k > 0$. The sequence of sampling instants $\{t_0,\ t_1,\ ...,\ t_k\}$ is strictly increasing in the sense that $t_{k+1} > t_k$. It is obvious that the sampling intervals are bounded and lie in the set

$$[h_{min},\ h_{max}],\ h_{min}\ <\ h_{max}$$

3.6.6 Sampling Interval Selection

Sampling is a major process in a control system [143] where the sampling interval determines how frequently the sensors, controllers and actuator exchange their data through the shared communication networks. The sampling rate must be chosen properly to satisfy the Shannon's sampling theorem and also to minimize the data transmission. A higher sampling rate can improve performance and achieve higher disturbance rejection. However, this puts more data into the communication links, causes longer time delay and packet loss, and degrades system performance. Therefore there is an upper bound for the sampling period where the network traffic is saturated at this sampling period. Any sampling period smaller than that would cause longer time delay and packets losses, and degrades the performance in DecNCS system. The lower sampling interval bound can be estimated using schedulability conditions. Similar to NCS period tasks, in DecNCS system periodical tasks might be scheduled using rate-monotonic (RM) and Earliest Deadline First (EDF) algorithm. RM is a scheduling algorithm [121, 155] used in real-time operating systems with a static-priority scheduling class. The static priorities are assigned on the basis of the cycle duration of the job: the shorter the cycle duration means a higher job's priority. EDF or least time to go is a dynamic scheduling algorithm used in real-time operating systems. It places processes in a priority queue. Whenever a scheduling event occurs (task finishes, new task released, etc.) the queue will be searched for the process closest to its deadline. This process is the next to be scheduled for execution. EDF can guarantee that all deadlines are met provided that the total utilization is not more than 100%. So, compared to fixed priority scheduling techniques like RM scheduling, EDF can guarantee all the deadlines in the system at higher loading are achieved. Back to our sampling interval selection that will be shown as the following based on comparison between EDF and RM as the following:

$$\begin{aligned} \min(h) &= \max\Big(\frac{h+\tau_{rt}}{1}, \frac{h+\tau_{rt}}{n(2^{1/n}-1)}\Big) \\ &= \frac{h+\tau_{rt}}{n(2^{1/n}-1)} \cong \frac{\tau_{rt}}{0.69} \end{aligned} \tag{3.24}$$

where n is an integer number that represents the scheduled tasks that will be executed for control system over the network, for example. The denominator, that represents the maximum ratio of utilization to meet the sufficient schedulability, can be estimated by iterative approach and it found to be 0.693. In general, the sampling interval h is too small comparing to network delay and other delays like device processing time will be added if it is considered in the delays calculations.

3.6.7 Transmission Constraints

Since the plant and controller are communicating through a network, it is possible to have a type of network that allows one node to access the network and transmits its corresponding values at each sampling time. This will add constraints [49, 146] on the transmission, and we know that the actual input of the plant $\hat{u}_k$ is not equal to the controller output u_k and the actual input of the controller $\hat{y}_k$ is not equal to the plant output y_k due to network effects. In other words, we can say that $\hat{u}_k$ and $\hat{y}_k$ are networked versions of u_k and y_k, respectively or the noise corrupted signals. To explain the effect of transmission constraints, assume that the plant has N sensors and M actuators where only one node can send at a time, and then only the transmitted values will be updated, while other values remain unchanged. This means that the constrained data exchange can be expressed as the following:

$$\begin{aligned} \hat{u}_k^i &= \Gamma_{\sigma_k}^u u_k + (1 - \Gamma_{\sigma_k}^u)\hat{u}_{k-1} \\ \hat{y}_k^i &= \Gamma_{\sigma_k}^y y_k + (1 - \Gamma_{\sigma_k}^y)\hat{y}_{k-1} \end{aligned} \tag{3.25}$$

where $\sigma_k = 1,\ ...,\ M + N,$ is used as switched function to determine which node will have the access to transmit, Γ_ℓ^u and Γ_ℓ^y are diagonal matrices where the j^{th} diagonal value is 1 if input/output belongs to node ℓ and zero otherwise. For example if system 3 is only allowed to send output measurements out of 5 systems in the network, then $\Gamma_{\sigma_k}^y$ will be:

$$\Gamma^{y}_{\sigma_k} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{3.26}$$

3.7 Control Design

In DecNCS we may not have all state space variables available for measurements and it could be not practical to measure all of them or it is too expensive to measure all state space variables. In order to be able to apply the state feedback control to a system, all of its state space variables must be available at all times. One of the solutions can be achieved by estimating system state space variables. This can be done by constructing another dynamical system called the observer or estimator, connected to the system under consideration, whose role is to produce good estimates of the state space variables of the original system. Since the whole sate vector is not available for feedback, then we can measure only $y = Cx$ from linear systems, a simple state-feedback control law would be $u = Kx$, where the gain K is chosen so that the closed-loop matrix $(A+BK)$ is stable. In practice, we cannot measure x, so that we have to use an output feedback design. Static output feedback design, that is $u = Ky$ turns out to be relatively hard to solve and does not guarantee closed-loop stability. The most common and systematic approach is to use a dynamic output feedback, where the controller (or compensator) has its own dynamics. The simplest form is an observer structure

$$\begin{aligned}\dot{\tilde{x}}_i &= A_i\tilde{x}_i + B_iu_i + L_i(y_i - C_i\tilde{x}_i)\\ u_i &= -K_i\tilde{x}_i, \quad i = 1,\ ...,\ n \end{aligned} \tag{3.27}$$

In this simple approach, $\tilde{x}_i$ is an estimate for the actual x for each subsystem i and we need to pick a good observation gain L_i such that $\tilde{x}_i \to x$ as fast as possible. In this work we will use decentralized observer-based controllers where the controllers can exchange information and have information about external states. Furthermore, the model-based controllers will adopt switching gains to deal with the transmission constraints effectively. The i^{th} networked observer-based controller is given by considering all network side effects we have discussed in this work:

$$\mathcal{C}^i_{\sigma_k} = \begin{cases} \tilde{x}^i_{k+1} = A^i_p\tilde{x}^i_k + B^i_{p1}\hat{u}^i_k + B^i_{p2}\hat{u}^i_{k-1} + L^i_{\sigma_k}\Gamma^y_{i,\sigma_k}(\hat{y}^i_k - C^i_d\tilde{x}^i_k) + \sum_{j=1,j\neq i}^{N} A^{i,j}_p\hat{x}^j_k \\ \hat{u}^i_k \quad = \ - K^i_{\sigma_k}\ \tilde{x}^i_k \end{cases} \tag{3.28}$$

where $\tilde{x}^i_{k+1}$ represents the state estimate at time $k+1$ for the plant state x^i_{k+1}, $B^i_{p1} = [\int_0^{h-\tau^k_{rt}} e^{As}ds]B$ and $B^i_{p2} = [\int_{h-\tau^k_{rt}}^{h} e^{As}ds]B$ when $\tau^k_{rt} \leq h$.

The output related matrices $L^i_{\sigma_k},\ K^i_{\sigma_k},\ i = 1,\ ...,\ N$ are the subsystem gain matrices. The state estimation error is

$$\psi^i_k \ = \ \tilde{x}^i_k - x^i_k \tag{3.29}$$

The dynamics of all controllers can be shown in discrete model that is composed of block diagonal matrices due to the decentralized nature of the controllers and the same [49] will be rewritten without the superscript i.

3.8 Closed Loop System

3.8.1 Introduction

In control engineering, it is very common to hear of the two terminologies namely "**Open-Loop Control**" and "**Closed-Loop Control**". In Open-Loop control no feedback loop is employed and system variations which cause the output to deviate from the desired value are not detected or corrected. A Closed-Loop system utilizes feedback to measure the actual system operating parameter being controlled such as temperature, pressure, flow, level, or speed. This feedback signal is sent back to the controller where it is compared with the desired system setpoint. The controller develops an error signal that initiates corrective action and drives the final output device to the desired value. In the DC Motor Drive illustrated above, the tachometer provides a feedback voltage which is proportional to the actual motor speed. Closed-Loop Systems have the following features:

- A **Reference** or Set Point that establishes the desired operating point around which the system controls.
- The **process variable** Feedback signal that tells the controller at what point the system is actually operating.
- A **Controller** which compares the system Reference with the system Feedback and generates an Error signal that represents the difference between the desired operating point and the actual system operating value.
- A **Final Control Element** or mechanism which responds to the system Error to bring the system into balance. This may be a pneumatically controlled valve, an electronic positioner, a positioning motor, an SCR or transistor power inverter, a heating element, or other control device.

- **System Tuning Elements** which modify the control operation by introducing mathematical constants that tailor the control to the specific application, provide system stabilization, and adjust system response time. In process control systems these tuning elements are: Proportional, Integral, and Derivative (PID) functions. For example, in electrical systems, such a generator voltage regulators and motor drives, typical tuning adjustments for such system may include:
  - **Gain**, the amplification factor of the controller error amplifier, which affects both system stability and response time.
  - **Stability** which provides a time-delayed response to feedback variations to prevent oscillations and reduce system hunting.
  - **Feedback and adjustment** which controls the amplitude of the feedback signal that is balanced against the system set-point.
  - **Boost** which is used in AC and DC motor drives to provide extra low-end torque.
  - **IR Compensation** which provides a control signal that compensates for the IR Drop (Voltage Drop) which occurs in the armature windings in DC machines due to increased current flow through the armature.

3.8.2 Derivations of Closed Loop System

To derive the closed-loop system, we will introduce the state vectors shown in eq. 3.30

$$\begin{aligned} \xi_k &:= \begin{bmatrix} x_k & \psi_k & e^u_k & e^y_k & d_k & w_k \end{bmatrix}^t \\ \xi_{k+1} &:= \begin{bmatrix} x_{k+1} & \psi_{k+1} & e^u_{k+1} & e^y_{k+1} & d_{k+1} & w_{k+1} \end{bmatrix}^t \end{aligned} \tag{3.30}$$

where

$$d_k^i \ := \ \sum_{j=1,j\neq i}^{N} A_p^{i,j}\hat{x}_k^j, \ \ w_k^i := \sum_{j=1,j\neq i}^{N} C_{i,j}\hat{x}_k^j \tag{3.31}$$

and combining the foregoing relations, the overall closed-loop dynamics can be expressed as

$$\xi_{k+1} \ = \ \mathcal{A}_{cl}\,\xi_k \tag{3.32}$$

where the closed loop matrix is shown in eq. 3.33.

$$\mathcal{A}_{cl} \ = \ \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & a_{15} & 0 \\ 0 & a_{22} & 0 & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & a_{35} & a_{36} \\ a_{41} & a_{42} & a_{43} & a_{44} & 0 & a_{46} \\ 0 & 0 & 0 & 0 & a_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{66} \end{bmatrix} \tag{3.33}$$

$$\begin{aligned}
a_{11} &= A_p^i - B_p^i K_{\sigma_k}^i, \;\; a_{12} = -B_p^i K_{\sigma_k}^i, \\
a_{13} &= -B_p^i (I - \alpha_k \Gamma_{\sigma_k}^u), \;\; a_{15} = I, \\
a_{22} &= A_p^i - \beta_k L_{\sigma_k}^i \Gamma_{i,\sigma_k}^y C_p^i, \\
a_{31} &= (A_p^i - K_{\sigma_k}^i B_p^i - I) K_{\sigma_k}^i, \\
a_{32} &= K_{\sigma_k}^i A_p^i - K_{\sigma_k}^i (I - B_p^i K_{\sigma_k}^i) - K_{\sigma_k}^i L_{\sigma_k}^i \Gamma_{i,\sigma_k}^y C_p^i \;\; , \\
a_{33} &= (K_{\sigma_k}^i B_p^i + I)(I - \alpha_k \Gamma_{\sigma_k}^u) \;\; , \\
a_{35} &= K_{\sigma_k}^i \;\; , a_{36} = (I + K_{\sigma_k}^i B_p^i) \;\; , \\
a_{41} &= C_p^i (-A_p^i + \alpha_k K_{\sigma_k}^i B_p^i + I), \\
a_{42} &= \alpha_k K_{\sigma_k}^i B_p^i C_p^i, \;\; a_{43} = C_p^i B_p^i (I - \alpha_k \Gamma_{i,\sigma_k}^u), \\
a_{44} &= (I - \beta_k \Gamma_{i,\sigma_k}^y), \\
a_{55} &= \Theta \;\; , a_{66} = \Psi \qquad (3.34)
\end{aligned}$$

However, if we seek a fully decentralized structure in which case the exchange of state information among subsystems is not allowed ($d_k = 0$, $w_k = 0$), then (3.30)-(3.34) will be reduced to:

$$\begin{aligned}
\zeta_k &= \begin{bmatrix} x_k & \psi_k & e_k^u & e_k^y \end{bmatrix} \qquad (3.35) \\
\zeta_{k+1} &= \mathcal{A}_{de} \, \zeta_k \\
\mathcal{A}_{de} &= \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \qquad (3.36)
\end{aligned}$$

where a_{11}, ..., a_{44} are given by (3.34). The design complexity can be further reduced by assuming that each controller has the accessibility to send the control signals to actuators at any time instant, hence there is no constraint on sending but still there is constraints on y. This implies that $e^u_k = 0$,$\Gamma^u_{i,\sigma_k} = I$ As a result, the closed-loop system matrix in (3.36) will be reduced to

$$\mathcal{A}_{dd} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.37}$$

Proof. This proof demonstrate in details the derivations of A_{cl} in 3.32 and 3.30 and for simplicity, we keep A_p as A only in writing the proof. Note that we have used the equations (3.24,3.25,3.21,3.27) while deriving the proofs. Also, we will use the derivations of equations (3.38,3.39) for the proofs of equations (3.40-3.44).

$$\begin{aligned}
\hat{u}^i_k &= \alpha\Gamma^u_{i,\sigma_k} u^i_k + (1-\alpha\Gamma^u_{i,\sigma_k})\hat{u}^i_{k-1} \\
\hat{u}^i_k &= \alpha\Gamma^u_{i,\sigma_k} u^i_k + (1-\alpha\Gamma^u_{i,\sigma_k})[e^u_k + u^i_k] \\
\hat{u}^i_k &= u^i_k + (1-\alpha\Gamma^u_{i,\sigma_k})e^u_k \\
\hat{u}^i_k &= -K_i\tilde{x}^i_k + (1-\alpha\Gamma^u_{i,\sigma_k})e^u_k \\
\hat{u}^i_k &= -K_i(\psi^i_k + x^i_k) + (1-\alpha\Gamma^u_{i,\sigma_k})e^u_k \\
\hat{u}^i_k &= -K_i\psi^i_k - K_i x^i_k + (1-\alpha\Gamma^u_{i,\sigma_k})e^u_k
\end{aligned} \tag{3.38}$$

$$\begin{aligned}
(\hat{y}_k^i - \tilde{y}_k^i) &= \alpha_k \Gamma_{i,\sigma_k}^u u_k^i + (1 - \alpha_k \Gamma_{i,\sigma_k}^u) \hat{u}_{k-1}^i \\
(\hat{y}_k^i - \tilde{y}_k^i) &= \alpha_k \Gamma_{i,\sigma_k}^u u_k^i + (1 - \alpha_k \Gamma_{i,\sigma_k}^u)[e_k^u + u_k^i] \\
(\hat{y}_k^i - \tilde{y}_k^i) &= u_k^i + (1 - \alpha \Gamma_{i,\sigma_k}^u) e_k^u \\
(\hat{y}_k^i - \tilde{y}_k^i) &= -K_i \tilde{x}_k^i + (1 - \alpha_k \Gamma_{i,\sigma_k}^u) e_k^u \\
(\hat{y}_k^i - \tilde{y}_k^i) &= -K_i (\psi_k^i + x_k^i) + (1 - \alpha_k \Gamma_{i,\sigma_k}^u) e_k^u \\
(\hat{y}_k^i - \tilde{y}_k^i) &= -K_i \psi_k^i - K_i x_k^i + (1 - \alpha_k \Gamma_{i,\sigma_k}^u) e_k^u \qquad (3.39)
\end{aligned}$$

The work shown in eq. 3.40 is for the first parameter in eq. 3.30 which the state at time $k+1$ for system i.

$$\begin{aligned}
x_{k+1}^i &= A_i x_k^i + B_i \hat{u}_k^i \\
x_{k+1}^i &= A_i x_k^i + B_i (\alpha_k \Gamma_{i,\sigma_k}^u u_k^i + (1 - \alpha_k \Gamma_{i,\sigma_k}^u) \hat{u}_{k-1}^i) \\
x_{k+1}^i &= A_i x_k^i + \alpha_k \Gamma_{i,\sigma_k}^u B_i u_k^i + (1 - \alpha_k \Gamma_{i,\sigma_k}^u) B_i [e_k^u + u_k^i] \\
x_{k+1}^i &= A_i x_k^i + \alpha_k \Gamma_{i,\sigma_k}^u B_i u_k^i + B_i u_k^i + B_i e_k^u - \alpha_k \Gamma_{i,\sigma_k}^u B_i e_k^u - \alpha_k \Gamma_{i,\sigma_k}^u B_i u_k^i \\
x_{k+1}^i &= A_i x_k^i + B_i u_k^i + B_i e_k^u - \alpha_k \Gamma_{i,\sigma_k}^u B_i e_k^u \\
x_{k+1}^i &= A_i x_k^i + B_i (-K_i \tilde{x}_k^i) + [I - \alpha_k \Gamma_{i,\sigma_k}^u] B_i e_k^u \\
x_{k+1}^i &= A_i x_k^i - B_i K_i x_k^i - B_i K_i \psi_k^i + [I - \alpha_k \Gamma_{i,\sigma_k}^u] B_i e_k^u \\
x_{k+1}^i &= [A_i - B_i K_i] x_k^i - B_i K \psi_k^i + [I - \alpha_k \Gamma_{i,\sigma_k}^u] B_i e_k^u \qquad (3.40)
\end{aligned}$$

Then, the work shown in eq. 3.41 is for the first parameter in eq. 3.30 which the state at estimation error at time $k+1$ for system i.

$$\begin{aligned}
\psi^i_{k+1} &= \tilde{x}^i_{k+1} - x^i_{k+1} \\
\psi^i_{k+1} &= A_i\tilde{x}^i_k + B_i\hat{u}^i_k + \beta_k L_i \Gamma^y_{i,\sigma_k}(\hat{y}^i_k - \tilde{y}^i_k) - (A_i x^i_k + B_i\hat{u}^i_k) \\
\psi^i_{k+1} &= A_i(\psi^i_k + x^i_k) + B_i\tilde{u}^i_k + \beta_k L_i \Gamma^y_{i,\sigma_k}(\hat{y}^i_k - \tilde{y}^i_k) - A_i x^i_k - B_i\hat{u}^i_k \\
\psi^i_{k+1} &= A_i\psi^i_k + \beta_k L_i \Gamma^y_{i,\sigma_k}(\hat{y}^i_k - \tilde{y}^i_k) \\
\psi^i_{k+1} &= A_i\psi^i_k + \beta_k L_i \Gamma^y_{i,\sigma_k}(\hat{y}^i_k - C_i(\psi^i_k + x^i_k)) \\
\psi^i_{k+1} &= A_i\psi^i_k + \beta_k L_i \Gamma^y_{i,\sigma_k}(C_i x^i_k - C_i\psi^i_k - C_i x^i_k) \\
\psi^i_{k+1} &= (A_i - \beta_k L_i \Gamma^y_{i,\sigma_k} C_i)\psi^i_k \qquad (3.41)
\end{aligned}$$

After that, we show the derivation of the 3^{rd} parameter which is the communication constraints on u at time $k+1$ for system i.

$$\begin{aligned}
e^u_{k+1} &= \hat{u}^i_k - u^i_{k+1} \\
e^u_{k+1} &= \hat{u}^i_k - (-K_i\tilde{x}^i_{k+1}) \\
e^u_{k+1} &= \hat{u}^i_k + K_i(A_i\tilde{x}^i_k + B_i\hat{u}^i_k + \beta_k L_i \Gamma^y_{i,\sigma_k}(\hat{y}^i_k - \tilde{y}^i_k)) \\
e^u_{k+1} &= \hat{u}^i_k + K_i A_i(\psi^i_k + x^i_k) + K_i B_i\hat{u}^i_k + K_i\beta_k L_i \Gamma^y_{i,\sigma_k}(\hat{y}^i_k - \tilde{y}^i_k) \\
e^u_{k+1} &= -K_i\psi^i_k - K_i x^i_k + (1 - \alpha_k\Gamma^u_{i,\sigma_k})e^u_k + K_i A_i(\psi^i_k + x^i_k) + K_i B_i\hat{u}^i_k - K_i\beta_k L_i C_i\psi^i_k\Gamma^y_{i,\sigma_k} \\
e^u_{k+1} &= R_1\psi^i_k + R_2 x^i_k + R_3 e^u_k + K_i d_k + (I + K_i B_i)w_k \qquad (3.42)
\end{aligned}$$

where

$$\begin{aligned} R_1 &= [-(I + K_iB_i)K_i + K_iA_i - K_iL_i\beta\Gamma^y_{i,\sigma_k}] \\ R_2 &= [-(I + K_iB_i) + K_iB_i] \\ R_3 &= (I + K_iB_i)(1 - \alpha_k\Gamma^u_{i,\sigma_k}) \end{aligned} \tag{3.43}$$

Finally, we show the derivation of the 4^{th} parameter which is the communication constraints on y at time $k+1$ for system i.

$$\begin{aligned} e^y_{k+1} &= \hat{y}^i_k - \tilde{y}^i_k \\ e^y_{k+1} &= \beta\Gamma^y_{i,\sigma_k}y^i_k + (I - \beta\Gamma^y_{i,\sigma_k})\hat{y}^i_{k-1} - C_ix^i_{k+1} \\ e^y_{k+1} &= \beta\Gamma^y_{i,\sigma_k}y^i_k + (I - \beta\Gamma^y_{i,\sigma_k})[e^y_k + y^i_k] - C_i(A_ix^i_k + B_i\hat{u}^i_k) \\ e^y_{k+1} &= \beta\Gamma^y_{i,\sigma_k}y^i_k + (I - \beta\Gamma^y_{i,\sigma_k})[e^y_k + y^i_k] - C_iA_ix^i_k - C_iB_i\hat{u}^i_k \\ e^y_{k+1} &= (I - \beta\Gamma^y_{i,\sigma_k})e^y_k + y^i_k - C_iA_ix^i_k - C_iB_i[-K_ix^i_k - K\psi^i_k + (I - \alpha\Gamma^u_{i,\sigma_k})] \\ e^y_{k+1} &= (I - \beta\Gamma^y_{i,\sigma_k})e^y_k + C_ix^i_k - C_iA_ix^i_k + C_iB_iK_ix^i_k + C_iB_iK_i\psi^i_k - C_iB_i(I - \alpha_k\Gamma^u_{i,\sigma_k}) \\ e^y_{k+1} &= (I - \beta\Gamma^y_{i,\sigma_k})e^y_k + C_i(I - A_i - B_iK_i)x^i_k + C_iB_iK_i\psi^i_k - C_iB_i(I - \alpha_k\Gamma^u_{i,\sigma_k}) \end{aligned} \tag{3.44}$$

∎

3.9 Stability Analysis

The stability of a control system is often extremely important and is generally a safety issue in the engineering of a system. An example to illustrate the importance of stability is the control of a nuclear reactor. Also, more examples can include the chemical reactor which must maintain a stable flow with certain

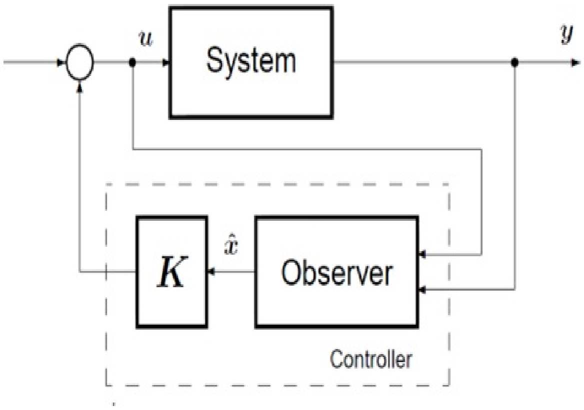


Figure 3.8: Closed-loop observer-based control system

pressure and temperature otherwise it will explode. An instability of this system could result in an unimaginable catastrophe. The stability of a system relates to its response to inputs or disturbances. A system which remains in a constant state unless affected by an external action and which returns to a constant state when the external action is removed can be considered to be stable. Control analysis is concerned not only with the stability of a system but also the degree of stability of a system. To know that the system is stable is not generally sufficient for the requirements of control system design. There is a need for stability analysis to determine how close the system is to instability and how much stability margin does it have when disturbances are present and when the gain is adjusted.

Remark 3.1 *Note that the observer has the same structure as the system plus the driving feedback term. The latter is sent over a communication network, that contains information about the observation error. The role of the feedback term is to reduce the observation error to zero (at steady state). Fig. 3.8 shows the ideal observer before introducing the network and Fig. 3.9 gives a networked controller where the observer-based elements are placed in the feedback loop.*

In view of the block-diagonal structure, let $\mathcal{A}_{cl} = blockdiag\{\mathcal{A}^t_{1,cl},\ ...,\ \mathcal{A}^t_{N,cl}\}$, we have the following as preliminary result:

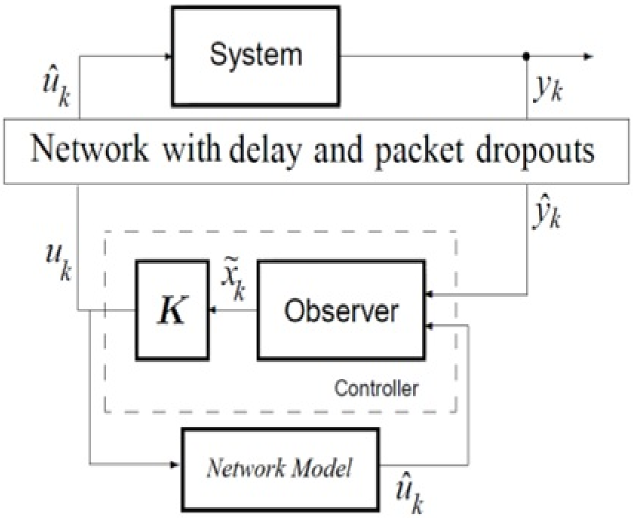


Figure 3.9: Closed-loop observer-based networked control system

Lemma 3.1 *The matrix inequality*

$$-\mathcal{M} + \mathcal{N}\,\Omega^{-1}\,\mathcal{N}^t < 0 \tag{3.45}$$

holds for some $0 < \Omega = \Omega^t \in \Re^{n\times n}$, *if and only if*

$$\begin{bmatrix} -\mathcal{M} & \mathcal{N}\mathcal{X} \\ \bullet & -\mathcal{X} - \mathcal{X}^t + \mathcal{Z} \end{bmatrix} < 0 \tag{3.46}$$

holds for some matrices $\mathcal{X} \in \Re^{n\times n}$ *and* $\mathcal{Z} \in \Re^{n\times n}$.

Proof. ($\Longrightarrow$) By Schur complements, inequality 3.45 is equivalent to

$$\begin{bmatrix} -\mathcal{M} & \mathcal{N}\Omega^{-1} \\ \bullet & -\Omega^{-1} \end{bmatrix} < 0 \tag{3.47}$$

setting $\mathcal{X} = \mathcal{X}^t = \mathcal{Z} = \Omega^{-1}$, we readily obtain inequality 3.46. ($\Longleftarrow$) Since the matrix $[I \;\; \mathcal{N}]$ of full rank, we obtain

$$\begin{bmatrix} I \\ \mathcal{N}^t \end{bmatrix}^t \begin{bmatrix} -\mathcal{M} & \mathcal{N}\mathcal{X} \\ \bullet & -\mathcal{X} - \mathcal{X}^t + \mathcal{Z} \end{bmatrix} \begin{bmatrix} I \\ \mathcal{N}^t \end{bmatrix} < 0$$

$$-\mathcal{M} + \mathcal{N}\mathcal{Z}\mathcal{N}^t < 0 \Longleftrightarrow -\mathcal{M}+\mathcal{N}\Omega^{-1}\mathcal{N}^t < 0,\ \mathcal{Z} = \Omega^{-1}.$$

which completes the proof. ∎

Introduce $\mathcal{X} = blockdiag\{\mathcal{X}_1,\ ...,\ \mathcal{X}_N\}$ where the matrix $\mathcal{X}_i,\ i = 1, .., N$ has the following form:

$$\mathcal{X}_i \;=\; \begin{bmatrix} X_{i11} & X_{i12} & ... & ... & X_{i1n} \\ 0 & X_{i22} & X_{i23} & ... & X_{i2n} \\ 0 & 0 & X_{i33} & ... & X_{i3n} \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & X_{inn} \end{bmatrix} \tag{3.48}$$

We are in a position to present the main result:

Theorem 3.1 *The closed-loop system (3.33) is said to be asymptotically stable if there exists symmetric positive definite matrices* $0 < \mathcal{P}_i = \mathcal{P}_i^t \in \Re^{n_i \times n_i}$, $0 < \mathcal{X}_i \in \Re^{n_i \times n_i}$, $0 < \mathcal{Z}_i = \mathcal{Z}_i^t \in \Re^{n_i \times n_i}$, $\ i = 1,\ ...,\ N$ *and gain matrices* $K,\ L$ *such that the following LMIs*

$$\begin{bmatrix} -\mathcal{P}_i & \mathcal{A}_{i,cl}^t \mathcal{X}_i \\ \bullet & -\mathcal{X}_i - \mathcal{X}_i^t + \mathcal{Z}_i \end{bmatrix} \;<\; 0, \quad i = 1,\ ...,\ N \tag{3.49}$$

have a feasible solution for $i = 1,\ ...,\ N$

Proof. We define a global Lyapunov functional by

$$V = \xi_k^t \mathcal{P} \xi_k, \;\; \mathcal{P} = blockdiag\{\mathcal{P}_1, \; ..., \; \mathcal{P}_N\}, \; \mathcal{P}_i > 0 \tag{3.50}$$

Evaluating the first difference ΔV along the solutions of (3.32) yields

$$\Delta V = -\mathcal{P} + \mathcal{A}_{cl}^t \mathcal{P} \mathcal{A}_{cl} \tag{3.51}$$

According to Lyapunov stability theorem, a necessary and sufficient condition for stability is $V > 0, \; \Delta V < 0$, That $\mathcal{P}_i > 0$ implies that $V > 0$. Applying **Lemma** 3.1 to inequality $\Delta V < 0$ using (3.51) with $\mathcal{M} = \mathcal{P}_i, \; \mathcal{N} = \mathcal{A}_{i,cl}^t$ and invoking Schur complements, we readily obtain inequality(3.49). ∎

Remark 3.2 *It is significant to note in view of* **Lemma** *3.1 that the feedback gains in* **Theorem** *3.1 can be calculated from the direct LMI variables*

3.10 Controllers Gains

In order to find the controller gains K and L from the given LMI, we need to do backward substitutions and these derivations will be shown in the following.

First, we need to expand the term $\mathcal{A}_{i,cl}^t X_i$ and derive the equations that we will use to find the gains, but before that we will show the transpose of the closed loop matrix. Note that for simplicity in writing the equations we will use $A_p^i = A_i$, $K_{\sigma_k}^i = K_i$ and this applied to all similar matrices.

$$\mathcal{A}_{i,cl}^{t} = \begin{bmatrix} aa_{11} & 0 & aa_{31} & aa_{41} & 0 & 0 \\ aa_{12} & aa_{22} & aa_{32} & aa_{42} & 0 & 0 \\ aa_{13} & 0 & aa_{33} & aa_{43} & 0 & 0 \\ 0 & 0 & 0 & aa_{44} & 0 & 0 \\ aa_{15} & 0 & aa_{35} & 0 & aa_{55} & 0 \\ 0 & 0 & aa_{36} & a_{45} & 0 & aa_{66} \end{bmatrix} \tag{3.52}$$

where

$$\begin{aligned} aa_{11} &= (A_i - B_iK_i)^t = A_i^t - K_i^tB_i^t, \\ aa_{12} &= 0 \;\; , aa_{15} = 0 \;\; , aa_{16} = 0 \;\; , \\ aa_{13} &= (-(I + K_iB_i)K_i + K_iA_i)^t = -K_i^t(I + B_i^tK_i^t) + A_i^tK_i^t, \\ aa_{14} &= (C(I - A_i + B_iK_i))^t = (I - A_i^t + B_i^tK_i^t)C_i^t, \end{aligned} \tag{3.53}$$

$$\begin{aligned} aa_{21} &= -(B_iK_i)^t = -K_i^tB_i^t, \\ aa_{22} &= (A_i - \beta_kL_i\Gamma_{i,\sigma_k}^{y}C_i)^t = A_i^t - \beta_kC_i^t\Gamma_{i,\sigma_k}^{y}L_i^t, \\ aa_{23} &= (K_iA_i - K_i(I - B_iK_i) - \beta K_iL_i\Gamma_{i,\sigma_k}^{y}C_i)^t, \\ aa_{23} &= A_i^tK_i^t - (I + B_i^tK_i^t)K_i^t - \beta C_i^t\Gamma_{i,\sigma_k}^{y}L_i^tK_i^t, \\ aa_{24} &= (C_iB_iK_i)^t = K_i^tB_i^tC_i^t, \\ aa_{25} &= 0 \;\; aa_{26} = 0 \;\; , \end{aligned} \tag{3.54}$$

$$\begin{aligned}
aa_{31} &= (B_i(I-\alpha_k\Gamma^u_{i,\sigma_k}))^t = (I-\alpha_k\Gamma^u_{i,\sigma_k})^t B_i^t, \\
aa_{32} &= 0 \;\; aa_{35}=0 \;\; aa_{36}=0 \;\;, \\
aa_{33} &= ((I+K_iB_i)(I-\alpha_k\Gamma^u_{i,\sigma_k}))^t = (I-\alpha_k\Gamma^u_{i,\sigma_k})^t (I+K_iB_i)^t, \\
aa_{34} &= (-C_iB_i(I-\alpha_k\Gamma^u_{i,\sigma_k}))^t = -(I-\alpha_k\Gamma^u_{i,\sigma_k})^t B_i^t C_i^t,
\end{aligned} \tag{3.55}$$

$$\begin{aligned}
aa_{41} &= 0 \;\;, aa_{42}=0 \;\;, aa_{43}=0 \;\;, aa_{45}=0 \;\;, aa_{46}=0 \;\;, \\
aa_{44} &= (I-\beta_k\Gamma^y_{i,\sigma_k})^t,
\end{aligned} \tag{3.56}$$

$$\begin{aligned}
aa_{51} &= I \;\;, aa_{53}=K_i^t \;\;, aa_{55}=\Theta^t \;\;, \\
aa_{52} &= 0 \;\;, aa_{54}=0 \;\;, aa_{56}=0 \;\;,
\end{aligned} \tag{3.57}$$

$$\begin{aligned}
aa_{61} &= 0 \;\;, aa_{62}=0 \;\;, aa_{64}=0 \;\;, aa_{65}=0 \;\;, \\
aa_{63} &= I+B_i^tK_i^t \;\;, aa_{66}=\Psi^t
\end{aligned} \tag{3.58}$$

after applying the product of the $\mathcal{A}^t_{i,cl}X_i$ we will have another matrix named S which is composed of the following elements:

$$
\begin{aligned}
S_{11} &= aa_{11}X_{11} \quad , \\
S_{12} &= aa_{11}X_{12} + aa_{12}X_{22} \quad , \\
S_{13} &= aa_{11}X_{13} + aa_{12}X_{23} + aa_{13}X_{33} \quad , \\
S_{14} &= aa_{11}X_{14} + aa_{12}X_{24} + aa_{13}X_{34} + aa_{14}X_{44} \quad , \\
S_{15} &= aa_{11}X_{15} + aa_{12}X_{25} + aa_{13}X_{35} + aa_{14}X_{45} + aa_{15}X_{55} \quad , \\
S_{16} &= aa_{11}X_{16} + aa_{12}X_{26} + aa_{13}X_{36} + aa_{14}X_{46} + aa_{15}X_{56} + aa_{16}X_{66}
\end{aligned}
\tag{3.59}
$$

$$
\begin{aligned}
S_{21} &= aa_{21}X_{11} \quad , \\
S_{22} &= aa_{21}X_{12} + aa_{22}X_{22} \quad , \\
S_{23} &= aa_{21}X_{13} + aa_{22}X_{23} + aa_{23}X_{33} \quad , \\
S_{24} &= aa_{21}X_{14} + aa_{22}X_{24} + aa_{23}X_{34} + aa_{24}X_{44} \quad , \\
S_{25} &= aa_{21}X_{15} + aa_{22}X_{25} + aa_{23}X_{35} + aa_{24}X_{45} + aa_{25}X_{55} \quad , \\
S_{26} &= aa_{21}X_{16} + aa_{22}X_{26} + aa_{23}X_{36} + aa_{24}X_{46} + aa_{25}X_{55} + aa_{26}X_{66}
\end{aligned}
\tag{3.60}
$$

$$
\begin{aligned}
S_{31} &= aa_{31}X_{11} \quad , \\
S_{32} &= aa_{31}X_{12} + aa_{32}X_{22} \quad , \\
S_{33} &= aa_{31}X_{13} + aa_{32}X_{23} + aa_{33}X_{33} \quad , \\
S_{34} &= aa_{31}X_{14} + aa_{32}X_{24} + aa_{33}X_{34} + aa_{34}X_{44} \quad , \\
S_{35} &= aa_{31}X_{15} + aa_{32}X_{25} + aa_{33}X_{35} + aa_{34}X_{45} + aa_{35}X_{55} \quad , \\
S_{36} &= aa_{31}X_{16} + aa_{32}X_{26} + aa_{33}X_{36} + aa_{34}X_{46} + aa_{35}X_{55} + aa_{36}X_{66}
\end{aligned}
\tag{3.61}
$$

$$
\begin{aligned}
S_{41} &= aa_{41}X_{11} \quad , \\
S_{42} &= aa_{41}X_{12} + aa_{42}X_{22} \quad , \\
S_{43} &= aa_{41}X_{13} + aa_{42}X_{23} + aa_{43}X_{33} \quad , \\
S_{44} &= aa_{41}X_{14} + aa_{42}X_{24} + aa_{43}X_{34} + aa_{44}X_{44} \quad , \\
S_{45} &= aa_{41}X_{15} + aa_{42}X_{25} + aa_{43}X_{35} + aa_{44}X_{45} + aa_{45}X_{55} \quad , \\
S_{46} &= aa_{41}X_{16} + aa_{42}X_{26} + aa_{43}X_{36} + aa_{44}X_{46} + aa_{45}X_{55} + aa_{46}X_{66}
\end{aligned}
\tag{3.62}
$$

$$\begin{aligned}
S_{51} &= aa_{51}X_{11} \quad , \\
S_{52} &= aa_{51}X_{12} + aa_{52}X_{22} \quad , \\
S_{53} &= aa_{51}X_{13} + aa_{52}X_{23} + aa_{53}X_{33} \quad , \\
S_{54} &= aa_{51}X_{14} + aa_{52}X_{24} + aa_{53}X_{34} + aa_{54}X_{44} \quad , \\
S_{55} &= aa_{51}X_{15} + aa_{52}X_{25} + aa_{53}X_{35} + aa_{54}X_{45} + aa_{55}X_{55} \quad , \\
S_{56} &= aa_{51}X_{16} + aa_{52}X_{26} + aa_{53}X_{36} + aa_{54}X_{46} + aa_{55}X_{55} + aa_{56}X_{66}
\end{aligned} \tag{3.63}$$

$$\begin{aligned}
S_{61} &= aa_{61}X_{11} \quad , \\
S_{62} &= aa_{61}X_{12} + aa_{62}X_{22} \quad , \\
S_{63} &= aa_{61}X_{13} + aa_{62}X_{23} + aa_{63}X_{33} \quad , \\
S_{64} &= aa_{61}X_{14} + aa_{62}X_{24} + aa_{63}X_{34} + aa_{64}X_{44} \quad , \\
S_{65} &= aa_{61}X_{15} + aa_{62}X_{25} + aa_{63}X_{35} + aa_{64}X_{45} + aa_{65}X_{55} \quad , \\
S_{66} &= aa_{61}X_{16} + aa_{62}X_{26} + aa_{63}X_{36} + aa_{64}X_{46} + aa_{65}X_{55} + aa_{66}X_{66}
\end{aligned} \tag{3.64}$$

Note that we have some zero values for aa_{ij} in eq. 3.58 which will cancel some terms for S_{ij}. After substituting the values from eq. 3.58 for each term in S_{ij}, the eq. (3.10-3.10) will be as follows:

$$
\begin{aligned}
S_{11} &= (A_i^t - K_i^t B_i^t) X_{11} = A_i^t X_{11} - K_i^t B_i^t X_{11} \ , \\
S_{11} &= A_i^t X_{11} - Y_{11}^t \ , \\
Y_{11}^t &= K_i^t B_i^t X_{11} \Rightarrow K_i = (X_{11} B_i)^{-1} Y_{11} \ ,
\end{aligned}
\tag{3.65}
$$

$$
\begin{aligned}
S_{12} &= (A_i^t - K_i^t B_i^t) X_{12} = A_i^t X_{12} - K_i^t B_i^t X_{12} \qquad , \\
S_{12} &= A_i^t X_{12} - Y_{12}^t \ , \\
Y_{12}^t &= K_i^t B_i^t X_{12} \Rightarrow K_i = (X_{12} B_i)^{-1} Y_{12} \ ,
\end{aligned}
\tag{3.66}
$$

$$
\begin{aligned}
S_{13} &= (A_i^t - K_i^t B_i^t) X_{13} + (-K_i^t (I + B_i^t K_i^t) + A_i^t K_i^t) X_{33} \ , \\
S_{13} &= A_i^t X_{13} - K_i^t B_i^t X_{13} + [A_i^t K_i^t X_{33} - K_i^t B_i^t K_i^t X_{33} - K_i^t X_{33}] \ , \\
S_{13} &= A_i^t X_{13} - Y_{13}^t + [\Delta_{13}] \ , \\
Y_{13}^t &= K_i^t B_i^t X_{13} \Rightarrow K_i = (X_{13} B_i)^{-1} Y_{13} \ ,
\end{aligned}
\tag{3.67}
$$

$$
\begin{aligned}
S_{14} &= (A_i^t - K_i^t B_i^t) X_{14} + (-K_i^t (I + B_i^t K_i^t) + A_i^t K_i^t) X_{34} + (I - A_i^t + K_i^t B_i^t) C_i^t X_{44} \ , \\
S_{14} &= A_i^t X_{14} - K_i^t B_i^t X_{14} + [\Delta_{14}] + X_{44} - A_i^t X_{44} + K_i^t B_i^t C_i^t X_{44} \ , \\
S_{14} &= A_i^t X_{14} - Y_{14}^t + [\Delta_{14}] X_{34} + C_i^t X_{44} - A_i^t C_i^t X_{44} + M_{14}^t \ , \\
Y_{14}^t &= K_i^t B_i^t X_{14} \Rightarrow K_i = (X_{14} B_i)^{-1} Y_{14} \ , \\
M_{14}^t &= K_i^t B_i^t C_i^t X_{44} \Rightarrow K_i = (X_{44}^t C_i^t B_i^t)^{-1} M_{14},
\end{aligned}
\tag{3.68}
$$

$$
\begin{aligned}
S_{15} &= (A_i^t - K_i^t B_i^t) X_{15} + (-K_i^t (I + B_i^t K_i^t) + A_i^t K_i^t) X_{35} + (I - A_i^t + K_i^t B_i^t) C_i^t X_{45} \quad , \\
S_{15} &= A_i^t X_{15} - K_i^t B_i^t X_{15} + [\Delta_{15}] X_{35} + C_i^t X_{45} - A_i^t C_i^t X_{45} + K_i^t B_i^t C_i^t X_{45} \quad , \\
S_{15} &= A_i^t X_{15} - Y_{15}^t + [\Delta_{15}] X_{35} + C_i^t X_{45} - A_i^t C_i^t X_{45} + M_{15}^t \quad , \\
Y_{15}^t &= K_i^t B_i^t X_{15} \Rightarrow K_i = (X_{15} B_i)^{-1} Y_{15} \quad , \\
M_{15}^t &= K_i^t B_i^t C_i^t X_{45} \Rightarrow K_i = (X_{45}^t C_i^t B_i^t)^{-1} M_{15}, \qquad (3.69)
\end{aligned}
$$

$$
\begin{aligned}
S_{16} &= (A_i^t - K_i^t B_i^t) X_{16} + (-K_i^t (I + B_i^t K_i^t) + A_i^t K_i^t) X_{36} + (I - A_i^t + K_i^t B_i^t) C_i^t X_{46} \quad , \\
S_{16} &= A_i^t X_{16} - K_i^t B_i^t X_{16} + [\Delta_{16}] X_{36} + C_i^t X_{46} - A_i^t C_i^t X_{46} + K_i^t B_i^t C_i^t X_{46} \quad , \\
S_{16} &= A_i^t X_{16} - Y_{16}^t + [\Delta_{16}] X_{36} + C_i^t X_{46} - A_i^t C_i^t X_{45} + M_{16}^t \quad , \\
Y_{16}^t &= K_i^t B_i^t X_{16} \Rightarrow K_i = (X_{16} B_i)^{-1} Y_{16} \quad , \\
M_{16}^t &= K_i^t B_i^t C_i^t X_{46} \Rightarrow K_i = (X_{46}^t C_i^t B_i^t)^{-1} M_{16}, \qquad (3.70)
\end{aligned}
$$

An important note that we assume that all matrices are **invertible**, **stabilizable** and **detectable**. Also, we can see from the eqs. (3.65 - 3.70) the repetition of K_i and L_i equivalent equations, so we will either eliminate some terms and make them zero or make them equal to each other. Using that we will have $X_{12} = X_{13} = X_{14} = X_{15} = X_{16} = X_{11}$ or make them all zero except X_{11}.

$$
S_{21} = -K_i^t B_i^t X_{11} = -Y_{11}^t \quad , \qquad (3.71)
$$

$$\begin{aligned}
S_{22} &= -K_i^t B_i^t X_{12} + (A_i^t - \beta_k C_i^t \Gamma^y_{i,\sigma_k})^t L_i^t X_{22}, \\
S_{22} &= -Y_{12}^t + A_i^t X_{22} - \beta_k Z_{22}^t \ \ , \\
Z_{22}^t &= C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t X_{22} \Rightarrow L_i = X_{22}^{-1} Z_{22} (\Gamma^y_{i,\sigma_k} C_i)^{-1} \qquad (3.72)
\end{aligned}$$

$$\begin{aligned}
S_{23} &= -K_i^t B_i^t X_{13} + (A_i^t - \beta C_i^t \Gamma^y_{i,\sigma_k})^t L_i^t X_{23} + [\Delta_{23}] X_{33} \ \ , \\
S_{23} &= -Y_{13}^t + A_i^t X_{23} - \beta_k Z_{23}^t \ \ , \\
\Delta_{23} &= K_i^t + K_i^t B_i^t K_i^t + A_i^t K_i^t - \beta C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t, \\
Z_{23}^t &= C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t X_{23} \Rightarrow L_i = X_{23}^{-1} Z_{23} (\Gamma^y_{i,\sigma_k} C_i)^{-1} \qquad (3.73)
\end{aligned}$$

$$\begin{aligned}
S_{24} &= -K_i^t B_i^t X_{14} + (A_i^t - \beta C_i^t \Gamma^y_{i,\sigma_k})^t L_i^t X_{24} + [\Delta_{34}] X_{34} + M_{14}^t \ \ , \\
S_{24} &= -Y_{14}^t + A_i^t X_{24} - \beta_k Z_{24}^t \ \ , \\
\Delta_{24} &= K_i^t + K_i^t B_i^t K_i^t + A_i^t K_i^t - \beta C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t, \\
Z_{24}^t &= C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t X_{24} \Rightarrow L_i = X_{24}^{-1} Z_{24}^t (\Gamma^y_{i,\sigma_k} C_i)^{-1} \qquad (3.74)
\end{aligned}$$

$$\begin{aligned}
S_{25} &= -K_i^t B_i^t X_{15} + (A_i^t - \beta C_i^t \Gamma^y_{i,\sigma_k})^t L_i^t X_{25} + [\Delta_{35}] X_{35} + M_{25}^t \ \ , \\
S_{25} &= -Y_{15}^t + A_i^t X_{25} - \beta_k Z_{25}^t \ \ , \\
\Delta_{25} &= K_i^t + K_i^t B_i^t K_i^t + A_i^t K_i^t - \beta C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t, \\
M_{25}^t &= K_i^t B_i^t C_i^t X_{25} \Rightarrow K_i = (X_{26}^t C_i^t B_i^t)^{-1} M_{25}, \\
Z_{25}^t &= C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t X_{25} \Rightarrow L_i = X_{25}^{-1} Z_{25}^t (\Gamma^y_{i,\sigma_k} C_i)^{-1} \qquad (3.75)
\end{aligned}$$

$$\begin{aligned}
S_{26} &= -K_i^t B_i^t X_{16} + (A_i^t - \beta C_i^t \Gamma^y_{i,\sigma_k})^t L_i^t X_{26} + [\Delta_{36}] X_{36} + M_{26}^t \quad , \\
S_{26} &= -Y_{16}^t + A_i^t X_{26} - \beta_k Z_{26}^t \quad , \\
\Delta_{26} &= K_i^t + K_i^t B_i^t K_i^t + A_i^t K_i^t - \beta C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t, \\
M_{26}^t &= K_i^t B_i^t C_i^t X_{26} \Rightarrow K_i = (X_{26}^t C_i^t B_i^t)^{-1} M_{26}, \\
Z_{26}^t &= C_i^t (\Gamma^y_{i,\sigma_k})^t L_i^t X_{26} \Rightarrow L = X_{26}^{-1} Z_{26}^t (\Gamma^y_{i,\sigma_k} C_i)^{-1}
\end{aligned} \tag{3.76}$$

$$\begin{aligned}
S_{31} &= (I - \alpha_k \Gamma^u_{i,\sigma_k})^t B_i^t X_{11} \quad , \\
S_{32} &= (I - \alpha_k \Gamma^u_{i,\sigma_k})^t B_i^t X_{12} \quad , \\
S_{33} &= (I - \alpha_k \Gamma^u_{i,\sigma_k})^t B_i^t X_{13} + [\Delta_1] X_{33} \quad , \\
S_{34} &= \Delta_1 X_{14} + \Delta_2 X_{34} + \Delta_3 X_{44} \quad , \\
S_{35} &= \Delta_1 X_{15} + \Delta_2 X_{35} + \Delta_3 X_{45} \quad , \\
S_{36} &= \Delta_1 X_{16} + \Delta_2 X_{36} + \Delta_3 X_{46} \quad , \\
\Delta_1 &= (I - \alpha_k \Gamma^u_{i,\sigma_k})^t (I - B_i^t K_i^t) \quad , \\
\Delta_2 &= (I - \alpha_k \Gamma^u_{i,\sigma_k}) (I + B_i^t K_i^t) \quad , \\
\Delta_3 &= -(I - \alpha_k \Gamma^u_{i,\sigma_k}) B_i^t C_i^t
\end{aligned} \tag{3.77}$$

$$\begin{aligned}
S_{41} &= 0 \quad , S_{42} = 0 \quad , S_{42} = 0 \quad , \\
S_{44} &= (I - \beta_k \Gamma^y_{i,\sigma_k}) X_{44} \quad , \\
S_{45} &= (I - \beta_k \Gamma^y_{i,\sigma_k}) X_{45} \quad , \\
S_{46} &= (I - \beta_k \Gamma^y_{i,\sigma_k}) X_{46}
\end{aligned} \tag{3.78}$$

$$
\begin{aligned}
S_{51} &= X_{11} \ , S_{52} = X_{12} \ , \\
S_{53} &= X_{13} + K_i^t X_{33} \ , \\
S_{54} &= X_{14} + K_i^t X_{34} \ , \\
S_{55} &= X_{15} + K_i^t X_{35} + \Theta^t X_{55} \ , \\
S_{56} &= X_{16} + K_i^t X_{36} + \Theta^t X_{56} \ , \qquad (3.79)
\end{aligned}
$$

$$
\begin{aligned}
S_{61} &= 0 \ , S_{62} = 0 \ , \\
S_{63} &= (I + B_i^t K_i^t) X_{33} \ , \\
S_{64} &= (I + B_i^t K_i^t) X_{34} \ , \\
S_{65} &= (I + B_i^t K_i^t) X_{35} \ , \\
S_{66} &= (I + B_i^t K_i^t) X_{36} + \Psi^t X_{35} \ , \qquad (3.80)
\end{aligned}
$$

the complexity work done in the previous equations for S matrix can be further reduced by assuming that each controller has the accessibility to send the control signals to actuators at any time instant, hence there is no constraint on sending. This implies that $e_k^u = 0$, $\Gamma_{i,\sigma_k}^u = I$, S_{3i} and all Δ_i will be eliminated, then S matrix elements will be rewritten as the following:

$$\begin{aligned}
S_{11} &= A_i^t X_{11} - Y_{11}^t \ , \\
S_{12} &= A_i^t X_{12} - Y_{12}^t \ , \\
S_{13} &= A_i^t X_{13} - Y_{13}^t \ , \\
S_{14} &= A_i^t X_{14} - Y_{14}^t + C_i^t X_{44} - A_i^t C_i^t X_{44} + M_{14}^t \ , \\
S_{15} &= A_i^t X_{15} - Y_{15}^t + C_i^t X_{45} - A_i^t C_i^t X_{45} + M_{15}^t \ , \\
S_{16} &= A_i^t X_{16} - Y_{16}^t + C_i^t X_{46} - A_i^t C_i^t X_{45} + M_{16}^t
\end{aligned} \tag{3.81}$$

$$\begin{aligned}
S_{21} &= -K_i^t B_i^t X_{11} = -Y_{11}^t \ , \\
S_{22} &= -Y_{12}^t + A_i^t X_{22} - \beta_k Z_{22}^t \ , \\
S_{23} &= -Y_{13}^t + A_i^t X_{23} - \beta_k Z_{23}^t \ , \\
S_{24} &= -Y_{14}^t + A_i^t X_{24} - \beta_k Z_{24}^t \ , \\
S_{25} &= -Y_{15}^t + A_i^t X_{25} - \beta_k Z_{25}^t \ , \\
S_{26} &= -Y_{16}^t + A_i^t X_{26} - \beta_k Z_{26}^t
\end{aligned} \tag{3.82}$$

$$\begin{aligned}
S_{41} &= 0 \ , S_{42} = 0 \ , S_{42} = 0 \ , \\
S_{44} &= (I - \beta_k \Gamma_{i,\sigma_k}^y) X_{44} \ , \\
S_{45} &= (I - \beta_k \Gamma_{i,\sigma_k}^y) X_{45} \ , \\
S_{46} &= (I - \beta_k \Gamma_{i,\sigma_k}^y) X_{46}
\end{aligned} \tag{3.83}$$

$$
\begin{aligned}
S_{51} &= X_{11} \ , S_{52} = X_{12} \ , \\
S_{53} &= X_{13} + K_i^t X_{33} \ , \\
S_{54} &= X_{14} + K_i^t X_{34} \ , \\
S_{55} &= X_{15} + K_i^t X_{35} + \Theta^t X_{55} \ , \\
S_{56} &= X_{16} + K_i^t X_{36} + \Theta^t X_{56}
\end{aligned} \tag{3.84}
$$

$$
\begin{aligned}
S_{61} &= 0 \ , S_{62} = 0 \ , \\
S_{63} &= 0 \ , S_{64} = 0 \ , S_{65} = 0 \ , \\
S_{66} &= \Psi^t X_{35}
\end{aligned} \tag{3.85}
$$

Finally, we have the following for the gains:

$$
\begin{aligned}
K_i &= (X_{11} B_i)^{-1} Y_{11} \ , \\
L_i &= X_{23}^{-1} Z_{23} (\Gamma^y_{i,\sigma_k} C_i)^{-1} \ ,
\end{aligned} \tag{3.86}
$$

3.11 Performance Measures

The performance of control system will be defined by how closely the system tracks a given reference trajectory. That is, given a desired reference trajectory $r(t)$ for the system, the performance measure is the difference between the actual system output $y(t)$ and the reference, $P = \| y - r \|$. Depending on the physical system and the application domain, one of many different norms may be used, including the maximum deviation from the trajectory, the average error along the trajectory, or the endpoint error. The baseline performance can be taken as the expected value of the performance criterion with no time delay. This characterization allows us to isolate the effect of the time delay or packet dropout or other network effects from the control design.

Consider a control system with and without time delay. Let $r(t)$ be the reference, $y^*(t)$ be the output of the system without time delay, and $y(t)$ be the output of the system with the time delay. The nominal performance criteria is given by:

$$P^* = \parallel y^* - r \parallel \tag{3.87}$$

We assume that the controller has been designed well, and that chosen performance criterion is the best possible performance that we can achieve. With time delay, the performance criteria becomes

$$\begin{aligned} P &= \parallel y - r \parallel \\ P &= \parallel y - y^* + y^* - r \parallel \\ P &\leq \parallel y - y^* \parallel + \parallel y^* - r \parallel \\ P &\leq \parallel \Theta \parallel + P^* \end{aligned} \tag{3.88}$$

where Θ represents the degradation in performance due to the network effects.

3.12 Simulation Studies

3.12.1 Chemical Reactors

In our everyday life we operate chemical processes, but we generally do not think of them in such a scientific fashion. Examples are running the washing machine or fertilizing our lawn. In order to quantify the efficiency of dirt removal in the washer, or the soil distribution pattern of our fertilizer, we need to know which transformation the chemicals will experience inside a defined volume, and how fast the transformation will be. Chemical kinetics and reactor engineering are the scientific foundation for the analysis of most environmental engineering processes, both occurring in nature and invented by men.

A good system that can be used as a benchmark system for system modeling, system identification, control, fault detection and diagnosis, as well as for fault-tolerant control is the Three-tanks system [208]. The system exhibits typical characteristics of a constrained hybrid system and has been proven useful to serve as a test bed for algorithms concerning state estimation, parameter identification, and control of hybrid systems. Here two configurations of the system are considered for the controller design . The three tank system is shown in Fig. 3.10. The system consists of three cylindrical tanks, $T1$, $T2$ and $T3$. $T1$ and $T2$ that are filled with liquid by two identical, independent pumps. The pumps deliver the liquid flows $Q1$ and $Q2$ and they can be continuously manipulated from a flow of 0 to a maximum flow $Qmax$. The tanks are interconnected to each other through pipes. The flow through these pipes can be interrupted with binary switching valves $V13$, $V23$ that can assume either the completely open or the completely closed position. The liquid levels $h1$, $h2$, $h3$, in each tank can be measured with level sensors. The nominal outflow from the system is located at the middle tank $T3$, i.e. $VL3$. The outflows $QL1$ and $QL2$ through valves $VL1$ and $VL2$ are zero

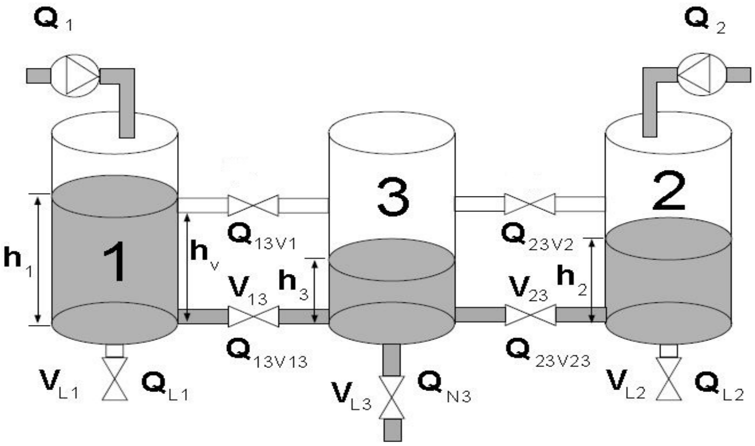


Figure 3.10: Three Tanks System

in nominal behavior and are used to model failures of the system. The system represents a chemical processing unit, with the outflow $QN3$ as the product. The overflow can be controlled by valves V_1 and V_2.

3.12.2 Simulated System

To illustrate the theoretical developments, we consider a plant as shown in the Fig. 3.11 where in such a system, we may have several control variables, states (tank level, inflow, temperature, outflow) and outputs that depends on the design requirements. The selected model will be used with the linearized system data as first numerical validation for the stability of the theorem proved in this work. We will not go in the details of how the system model was linearized, we only took the data from [171] that to be used for simulation.

The linearized model is described by the following matrices:

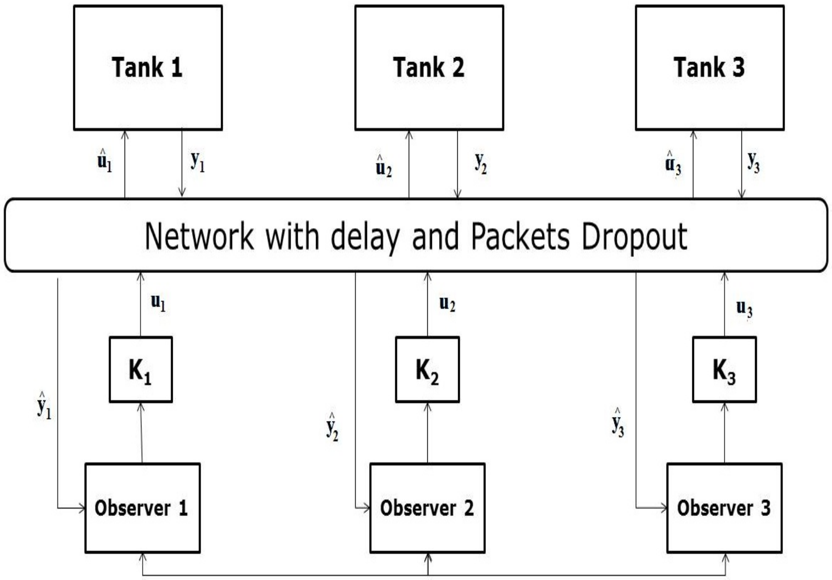


Figure 3.11: Closed-loop 3-Tank observer-based control system

$$\begin{aligned}
\mathcal{A}_j &= \begin{bmatrix} -a_{1j} & -1.01 & 0 & 0 \\ -3.2 & -a_{2j} & -12.8 & 0 \\ 6.4 & 0.347 & -a_{3j} & -1.04 \\ 0 & 0.833 & 11.0 & -a_{4j} \end{bmatrix}, \\
\mathcal{B}_j^t &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \\
\mathcal{A}_{dj} &= \begin{bmatrix} b_{1j} & 0 & 0 & 0 \\ 0 & b_{2j} & 0 & 0 \\ 0 & 0 & b_{3j} & 0 \\ 0 & 0 & 0 & b_{4j} \end{bmatrix}, \\
\mathcal{C}_j &= \begin{bmatrix} 10 & 0 & 0 & 0 \end{bmatrix}, \qquad (3.89)
\end{aligned}$$

where

$$
\begin{aligned}
a_{11} &= 4.931, \quad a_{12} = 4.886, \quad a_{13} = 4.902, \\
a_{21} &= 5.301, \quad a_{22} = 5.174, \quad a_{23} = 5.464, \\
a_{31} &= 35.511, \quad a_{32} = 30.645, \quad a_{33} = 31.773, \\
a_{41} &= 3.961, \quad a_{42} = 3.878, \quad a_{43} = 3.932, \\
b_{11} &= 1.921, \quad b_{12} = 1.915, \quad b_{13} = 1.908, \\
b_{21} &= 1.921, \quad b_{22} = 1.914, \quad b_{23} = 1.907, \\
b_{31} &= 1.878, \quad b_{32} = 1.866, \quad b_{33} = 1.869, \\
b_{41} &= 0.724, \quad b_{42} = 0.715, \quad b_{43} = 0.706
\end{aligned}
\tag{3.90}
$$

3.12.3 Simulation Results

Using the LMI toolbox in MATLAB, the observers gains for each subsystem are the following

$$
\begin{aligned}
L_1^t &= \begin{bmatrix} 0.0130 & 0.0681 & 0.0164 & -0.0385 \end{bmatrix}, \\
L_2^t &= \begin{bmatrix} -0.0158 & 0.1235 & 0.0162 & -0.0654 \end{bmatrix}, \\
L_3^t &= \begin{bmatrix} -0.0456 & 0.0701 & 0.0706 & -0.0201 \end{bmatrix}
\end{aligned}
$$

$$K_1 = \begin{bmatrix} 0.7573 & 0.7142 & 0.3973 & 0.8391 \\ 0.2138 & 8.2185 & 13.8882 & -3.4177 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} 0.3144 & -0.7983 & -3.8703 & 1.7806 \\ -0.6559 & 7.2776 & 14.8651 & -6.8895 \end{bmatrix},$$

$$K_3 = \begin{bmatrix} 0.2634 & -0.1587 & -3.1912 & 1.5713 \\ -1.0803 & 8.5794 & 12.2875 & -3.9658 \end{bmatrix}$$

As we mentioned earlier, P is positive definite and symmetric, see the following:

$$P_1 = \begin{bmatrix} 0.1448 & -0.0020 & 0.0005 & 0.0002 \\ -0.0020 & 0.1442 & -0.0005 & 0.0009 \\ 0.0005 & -0.0005 & 0.1420 & 0.0001 \\ 0.0002 & 0.0009 & 0.0001 & 0.1463 \end{bmatrix},$$

$$P_2 = \begin{bmatrix} 0.1022 & -0.0021 & 0.0006 & 0.0002 \\ -0.0021 & 0.1025 & -0.0005 & 0.0020 \\ 0.0006 & -0.0005 & 0.0993 & 0.0001 \\ 0.0002 & 0.0020 & 0.0001 & 0.1032 \end{bmatrix},$$

$$P_3 = \begin{bmatrix} 0.6502 & -0.0131 & 0.0042 & 0.0012 \\ -0.0131 & 0.6406 & -0.0032 & 0.0069 \\ 0.0042 & -0.0032 & 0.6280 & 0.0006 \\ 0.0012 & 0.0069 & 0.0006 & 0.6554 \end{bmatrix}$$

then we have X is positive definite,

$$\begin{aligned}
X_1 &= \begin{bmatrix} 0.2142 & -0.0560 & 0.0421 & -0.0153 \\ -0.0560 & 0.0383 & -0.0138 & 0.0515 \\ 0.0421 & -0.0138 & 0.0292 & 0.0435 \\ -0.0153 & 0.0515 & 0.0435 & 0.2597 \end{bmatrix}, \\
X_2 &= \begin{bmatrix} 0.1625 & -0.0200 & 0.0314 & -0.0291 \\ -0.0200 & 0.0275 & -0.0033 & 0.0672 \\ 0.0314 & -0.0033 & 0.0222 & 0.0341 \\ -0.0291 & 0.0672 & 0.0341 & 0.2809 \end{bmatrix}, \\
X_3 &= \begin{bmatrix} 0.1106 & -0.0172 & 0.0222 & -0.0131 \\ -0.0172 & 0.0144 & -0.0044 & 0.0257 \\ 0.0222 & -0.0044 & 0.0147 & 0.0207 \\ -0.0131 & 0.0257 & 0.0207 & 0.1264 \end{bmatrix}
\end{aligned}$$

$$
\begin{aligned}
Z_1 &= \begin{bmatrix} 0.1595 & -0.0419 & 0.0314 & -0.0118 \\ -0.0419 & 0.0284 & -0.0101 & 0.0387 \\ 0.0314 & -0.0101 & 0.0216 & 0.0325 \\ -0.0118 & 0.0387 & 0.0325 & 0.1943 \end{bmatrix}, \\
Z_2 &= \begin{bmatrix} 0.1115 & -0.0141 & 0.0214 & -0.0212 \\ -0.0141 & 0.0190 & -0.0020 & 0.0475 \\ 0.0214 & -0.0020 & 0.0151 & 0.0239 \\ -0.0212 & 0.0475 & 0.0239 & 0.1982 \end{bmatrix}, \\
Z_3 &= \begin{bmatrix} 0.7193 & -0.1135 & 0.1443 & -0.0893 \\ -0.1135 & 0.0925 & -0.0271 & 0.1686 \\ 0.1443 & -0.0271 & 0.0945 & 0.1350 \\ -0.0893 & 0.1686 & 0.1350 & 0.8279 \end{bmatrix}
\end{aligned}
$$

The following also shows the closed loop matrices.

$$Acl_1 = \begin{bmatrix} -5.8183 & -1.7242 & -0.3973 & -0.8391 \\ -4.0948 & -13.5195 & -26.6882 & -3.4177 \\ 6.2360 & 0.3470 & -32.5110 & -1.0400 \\ -0.3850 & 0.8330 & 11.0000 & -3.9610 \end{bmatrix},$$

$$Acl_2 = \begin{bmatrix} -5.3584 & -1.8083 & -3.8703 & -1.7806 \\ -5.0909 & -12.4516 & -27.6651 & -6.8895 \\ 6.2380 & 0.3470 & -30.6450 & -1.0400 \\ -0.6540 & 0.8330 & 11.0000 & -3.8780 \end{bmatrix},$$

$$Acl_3 = \begin{bmatrix} -5.6214 & -1.1687 & -3.1912 & -1.5713 \\ -4.9813 & -14.0434 & -25.0875 & -3.9658 \\ 5.6940 & 0.3470 & -31.7730 & -1.0400 \\ -0.2010 & 0.8330 & 11.0000 & -3.9320 \end{bmatrix}$$

Simulation of the closed-loop system is performed and the ensuing state trajectories are presented in Figs. (3.12-3.14). It is clearly evident that the interconnected systems are decentralized, asymptotically stabilizable with guaranteed performance.

3.13 Numerical Verification of the Solution

Since we are discussing a discrete time system, it will be good if we show some numerical proofs that support and validate the theory introduced in this chapter.

As we have stated in Lemma 3.1, that we should have some positive definite matrices (P, Z, X) and those are used for discrete system, then, the simple check to see wether the eign-values are less than one in magnitude or not.

By running this check in MATLAB code we will have the following that satisfy

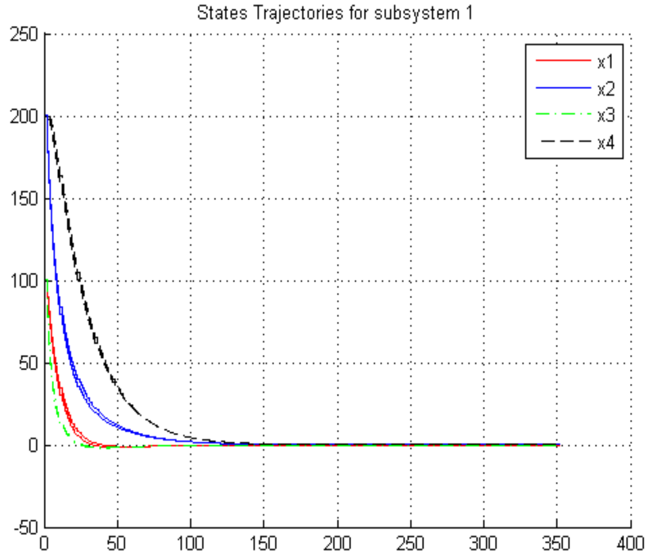


Figure 3.12: Closed-loop state trajectories: Subsystem 1

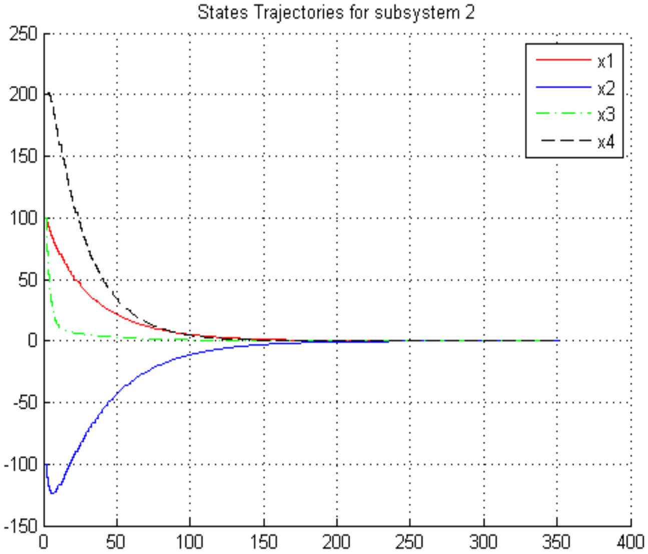


Figure 3.13: Closed-loop state trajectories: Subsystem 2

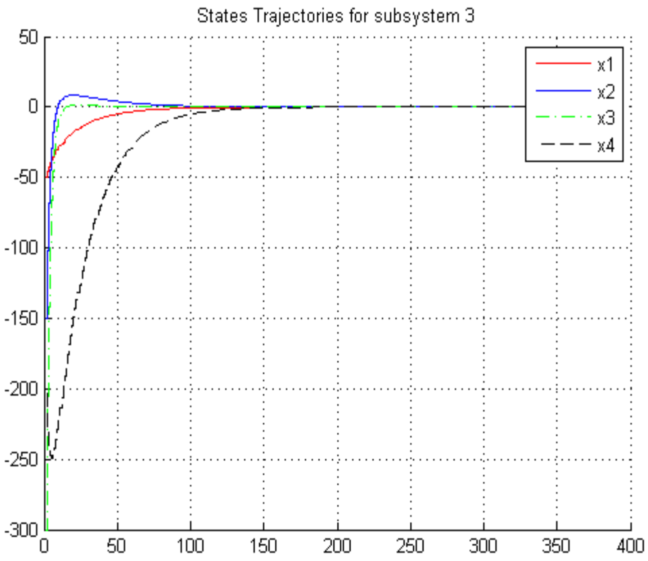


Figure 3.14: Closed-loop state trajectories: Subsystem 3

the above statement.

$$\begin{aligned} EigVal(X1) &= [0.0041, 0.0045, 0.0087, 0.0099] \\ EigVal(X2) &= [0.0188, 0.0194, 0.1039, 0.1478] \\ EigVal(X3) &= [0.1592, 0.2166, 0.2967, 0.3109] \qquad (3.91) \end{aligned}$$

$$\begin{aligned} EigVal(Z1) &= [0.0023, 0.0061, 0.0131, 0.0144] \\ EigVal(Z2) &= [0.0263, 0.0648, 0.1089, 0.1612] \\ EigVal(Z3) &= [0.2194, 0.2221, 0.6738, 0.9693] \qquad (3.92) \end{aligned}$$

$$\begin{aligned}
EigVal(P1) &= [0.0992, 0.0997, 0.1026, 0.1057] \\
EigVal(P2) &= [0.1419, 0.1424, 0.1458, 0.1473] \\
EigVal(P3) &= [0.6271, 0.6304, 0.6531, 0.6636] \qquad (3.93)
\end{aligned}$$

CHAPTER 4

A COMPARATIVE STUDY

4.1 Overview

In this chapter, different traffic light control structures over communication network, including the decentralized, quasi-decentralized and distributed networked strategies, are considered for coordinating and control of multiple intersections, which could be a great application of networked control signalized traffic light problem. It helps in achieving several objectives such as minimizing the waiting time during the red light period and perform better control in the next green cycle and more will be highlighted in this chapter. A state space model of traffic dynamics is proposed considering the effects of lossy communication network. Also, a sufficient condition for system stability is provided based on LMI. Finally, comparison and performance analysis of different types of networked control systems were done using simulation.

4.2 Introduction

In modern urban areas, the number of vehicles is growing larger and larger and the requirements for traveling by vehicles are becoming more demanding than ever.


Pollution Accidents Misbehavior Travel Delay

Figure 4.1: Traffic jam Consequences Examples

Even though large and sound traffic networks (freeways and roads) are already constructed, traffic congestion still cannot be avoided efficiently. It is often time and money consuming to build more common transportation infrastructures or reconstruct the ones that already exist. Therefore, traffic jams occur frequently and have a severe impact, when people need to use the common infrastructures with limited capacity at the same time, especially during rush hours [161]. Traffic congestion can give rise to traffic delays, economic losses, traffic pollution, and so on. To reduce traffic jams and to promote efficiency in traveling, effective traffic control methods are necessary. Several traffic control strategies were proposed and implemented in the field, like fuzzy control [139, 140, 183], PID, MPC and PLC control, to name a few. However, these algorithms are mainly focusing on controlling a single intersection or a single traffic control measure. These controllers are without global scope, and have limited control effect for the whole traffic network.

As we know, traffic intersections are not isolated; the traffic states of roads in a traffic network will interact with each other and a traffic jam that happens in one intersection may be caused by some irregular event (for example, an incident) that happened in another intersection in the same traffic network. Therefore, it is necessary to understand the behavior of traffic networks, and to investigate network-wide traffic coordinated control approaches that can coordinate and control traffic networks for a better performance. With respect to control systems, traditionally they utilize dedicated, point-to-point wired communication links us-

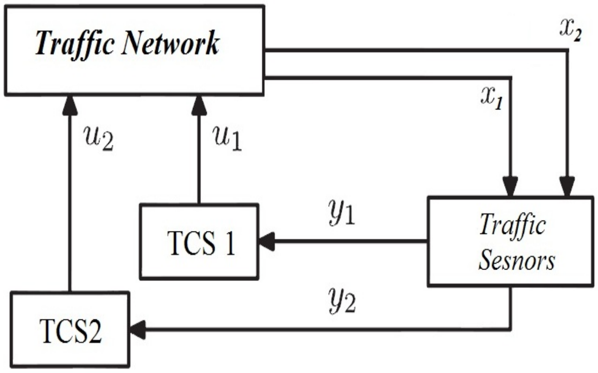


Figure 4.2: A Traditional Traffic Control System with two Control Loops

ing a small number of sensors and actuators to regulate appropriate process variables at desired values. The well known control strategy is the centralized control that has advantages for local intersection control but it has its own disadvantages for controlling the large traffic network with many signalized intersections.

For a system with multiple control loops, each intersection is a control loop and the controllers are designed to work in a decentralized fashion. Fig. 4.2 shows a traditional control system with two control loops, one for each intersection. The two traffic control systems (for example, TCS 1 and TCS 2) are designed based on two different continuously-sampled outputs, y_1 and y_2, of the system. The two controllers do not exchange information and operate in a decentralized fashion which makes each intersection isolated from the others. A similar system is shown in Fig. 4.3 but here it is over communication links, so it will be called Decentralized Networked Control Systems (DecNCS), and the red dashed lines represents the real-time network links. Communication networks make the transmission of data much easier and provide a higher degree of freedom in the configuration of control systems ([42]-[46]). Also, adding an additional information is easier using such a network where this information may be used to improve the closed-loop performance and the fault tolerance of a control system.

To coordinate multiple intersections across a long road we may need to consider

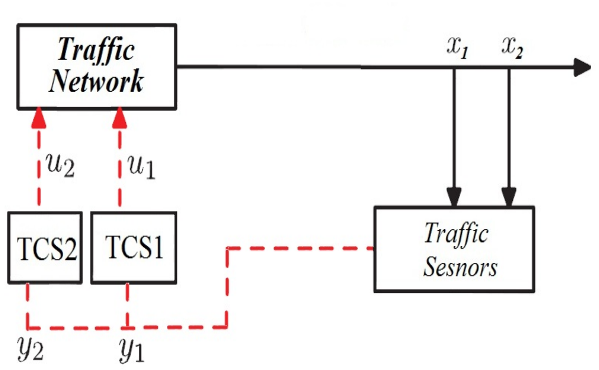


Figure 4.3: Decentralized Networked Traffic Control System

other techniques, namely, the Quasi-Decentralized and Distributed control. If we allow some exchange of information about the states between the intersections then we will have the quasi-decentralized model and with this we can for example exchange the information about vehicles queues. In the distributed model we may allow all types of information exchanges between intersections about the states and the executed control commands like the signal status whether it is red, green or yellow, the present queue in that signal before it become green, phase selection, phase timing and many other examples. The use of communication network (NCS) for these models will introduce the Quasi-DecNCS [39]-[40], (in our work we call it QuasiNCS) and distributed networked control (DNCS) [27]-[29] (see Fig.4.4). The hierarchical control structure [16], (see Fig.4.5) can be used also in case we are trying to control very large-scale traffic network with a large number of intersections. Instead of giving all the control authority to local controllers, the hierarchical control structure divides the control problem into multiple control problems at multiple levels.

All of these models can be used for the traffic signal intersection control to achieve a wide range of objective functions [165]-[169] and [200]-[203] such as:

1. Minimize overall delay to vehicles.
2. Minimize the waiting time at the intersection.

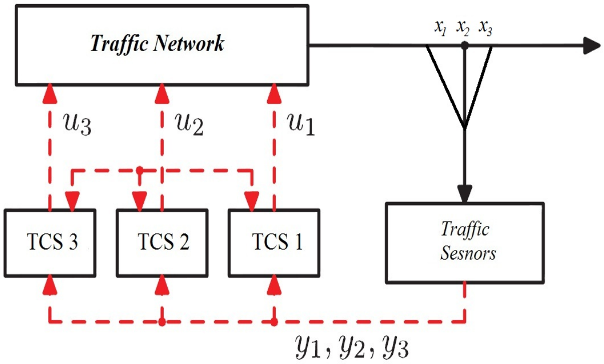


Figure 4.4: Distributed Networked Traffic Control System

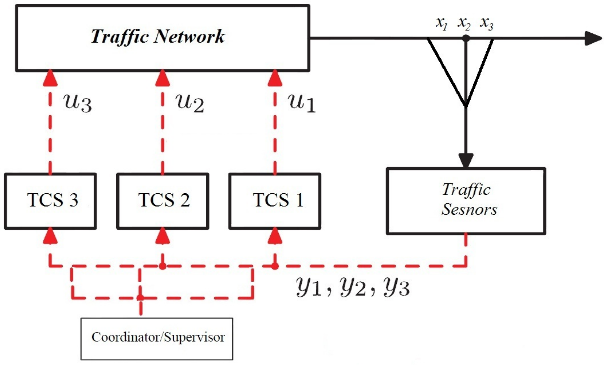


Figure 4.5: Hierarchical Networked Traffic Control System

3. Maximize the service time for each signal.
4. Minimize delays to public transport.
5. Minimize delays to emergency services.
6. Minimize delays to pedestrians.
7. Equitable distribution of delays between competing traffic.
8. Maximize reliability.
9. Maximize network capacity.
10. Minimize accident potential for all users.
11. Minimize environmental impact of vehicular traffic (noise, atmospheric pollution, visual intrusion)
12. Energy efficiency.
13. Handling of the red light crossing violations.
14. Transmission of traffic information to the police traffic control room.

It is important to note that some of the objectives do conflict and a compromise may have to be made in the selection of objectives. However, some objectives can be met in tandem, for example minimizing delay to vehicles would also help to minimize fuel consumption [204], atmospheric pollution and increase network throughput.

Generally, in planning and designing a traffic signal control system, one must first understand the applicable operational concepts related to signalized intersection control and signal-related special control. Signalized intersection control concepts include:

1. Isolated intersection control - controls traffic without considering adjacent signalized intersections.

2. Interchange and closely-spaced intersection control - provides progressive traffic flow through two closely spaced intersections, such as interchanges. Control is typically done with a single traffic controller.

3. Arterial intersection control (open network) - provides progressive traffic flow along the arterial intersection. This is accomplished by coordination of the traffic signals.

4. Closed network control - coordinates a group of adjacent signalized intersections.

5. Area-wide system control - treats all or a major portion of signals in a city (or metropolitan area) as a total system. Isolated, open- or closed-network concepts may control individual signals within this area

We may add also the signal-related special control concepts which includes:

1. High occupancy vehicle (HOV) priority systems.

2. Preemption - Signal preemption for emergency vehicles, railroads, and drawbridges.

3. Priority Systems - Traffic signal control strategies that assign priority for the movement of transit vehicles.

4. Directional controls - Special controls designed to permit unbalanced lane flow on surface streets and changeable lane controls.

5. Television monitoring.

6. Over height vehicle control systems.

Another important issue for traffic control system is how to collect the traffic data. This can be done by using Wireless Sensors Network (WSN), or wired sensor network, that will feed the control system with the number of incoming traffic data, passing vehicles and crossing the signals [166]. Many traffic light systems operate on a timing mechanism, preset cycle time that changes the lights after a given interval. An intelligent traffic light system senses the presence or absence of vehicles then it controls the traffic lights accordingly using one of the control approaches we have mentioned. The very obvious idea behind intelligent traffic systems is that drivers will not spend unnecessary time waiting for the traffic lights to change which may lead them to some traffic violations and accidents when some drivers start to lose their patience. An intelligent traffic system detects traffic in many different ways. The older system uses weight as a trigger mechanism. Current traffic systems react to motion to trigger the light changes based on the infrared object detector that picks up the presence of a car or some proximity switches. Then, a switch causes the lights to change. In order to accomplish this, algorithms are used to govern the actions of the traffic system [167, 168]. We need to understand the function of traffic signals so that we can improve driving habits by controlling the speed and the red light crossing in order to reduce the number of associated traffic accidents. The more the drivers know about the operation of traffic signals, the less frustrated they are going to be while waiting for the lights to change. Usually, in the intelligent traffic signal systems [190]-[192], the main aim is to reduce the cars waiting time at each signal and also to maximize the total number of cars that can cross an intersection safely during the green signal time.

Network-based communication allows for easy modification of the control strategy by rerouting signals, having redundant systems that can be activated automatically when component failure occurs, and in general, it allows having a high-level supervisory control over the entire system. However, augmenting existing control networks with real-time wired or wireless sensor and actuator networks challenges many of the assumptions made in the development of traditional process control methods dealing with dynamical systems linked through ideal channels with flawless, continuous communication. In the context of networked control systems, key issues that need to be carefully handled at the control system design level include data losses due to field interference and time delays due to network traffic as well as due to the potentially heterogeneous nature of the additional measurements. As a result, the controller will do a control over the network not through the network and design shall be robust to the following [42, 141]: uncertain time delays due communication, processing and queuing, transmission constraints where not all outputs and inputs can be transmitted at same time, quantization error, fixed and varying sampling intervals, sampling interval selection, unreliable transmission and packets dropout, network-induced errors and interconnected communication. The main aim is to design an intelligent controller, for traffic signal with multiple intersections, that can adapt to combined effects from these previously mentioned points, taken all together, which has not been done in the literature to achieve the objectives we mentioned.

4.3 Traffic Control Background and Related Works

A road intersection is a bottleneck point in the urban traffic network and it is a very critical node. Traffic may accumulate quickly and traffic jam can occur quickly in case the traffic control system is not efficient to properly manage the vehicles queues in a fast and smart manner. One of the hot topics these days is how to gather the traffic information and control the traffic flow around. There are conventional traffic light control methods like fix-time control [206], time of day control, vehicle actuated control, semi-actuated control, green wave control, area static control, area dynamic control and sensor-based control. Usually, the nature of traffic flow is random and predicting the traffic behavior is not easy. This section provides a survey of the literature related to traffic light control systems, highlighting most of the traffic light control models that were developed to improve traffic light efficiency and achieve several objectives as mentioned in the introduction.

1. Pre-timed control: all of the control parameters are fixed and preset off-line. Off-line techniques (for example, the various versions of the TRANSYT family of software packages are useful in generating the parameters for fixed timing plans for conventional pre-timed urban traffic control systems based on the deterministic traffic conditions during different time periods of the day (e.g., peak hours, off-peak hours).

2. Queue traffic light model (Simple, Extended , Event Driven). The queue length in each lane can be evaluated using different techniques depending on street width and the number of vehicles that are expected at a given time of day [179]-[181]. In this model, traffic light efficiency is effected when unexpected events happen (traffic accidents) causing disruption to the flow of vehicles. Extended queue model that is used to meet two objectives not only the queue length as in the simple queue model, also the waiting which is the time spent by the vehicle in the queue. For the event-driven, it is basically dependent on sensing device that sense like weighting cells so the priority will be given for that queue in that cycle.

3. Knowledge-based Models: Knowledge based systems are artificial intelligence tools that work in a narrow domain to provide intelligent decisions

with justification. Knowledge is acquired and represented using various knowledge representation rules, frames and scripts [176]-[178]. These models are designed to take advantage of the information and operational experience accumulated from previous traffic management experiences and incidents. For example, data can be stored like the detailed response time, incident duration, lane-blockage conditions, and the approximate traffic impacts on the network for each responded incident dealt with, ...etc. Such a knowledge base will offer the traffic control operators a reliable reference for estimating the potential impact due to a detected incident.Also, if we divide the network to problem areas or zones we can do the analysis of the situation using knowledge about traffic behavior and control criteria specific for that area.

4. Graph-based Models (Petri net Models): From the perspective of graph, one can transform a real traffic network into a graph in which vertices represent the intersections of roads, and edges represent the road segments. The Petri net Models consist of places (graphically represented as circles) and transitions (graphically represented as bars) connected via a set of directed arcs ([170]-[173]). Places may contain tokens (represented by dots inside the circle) that move through the network (for example, from place to place) according to certain rules. Petri net models have been used as a tool for various kinds of discrete event systems, simulation and control logic. This type of model has some disadvantages ([174]-[175]) and it is hard to manage.

5. Sensors-based Models: In these models different types of sensors can be used including wireless sensors. Examples of sensors can be inductive loop detectors, micro-loop probes, IR, LED, motion detectors and pneumatic road

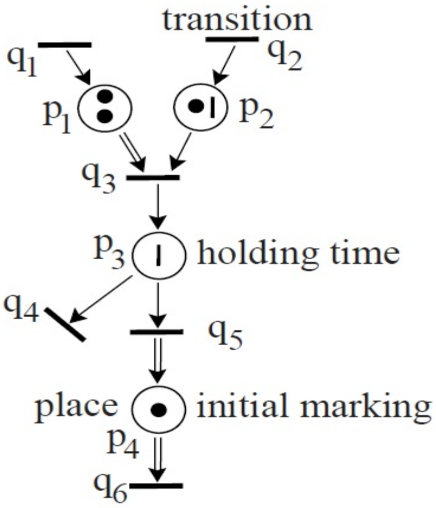


Figure 4.6: Petri Net Example

tubes. The sensor nodes count number of vehicles approaching an intersection and we can monitor each lane using sensors. The message sent from the sensor nodes to the intersection controller include several data such as the number of vehicles, time duration of the collected data, and lane number. According to the number of detection points, traffic forecasting algorithms can be classified as single-point, double-point and multi-point where the first one is the mostly used in the traditional traffic control. By using the wireless sensors network (WSN), there are several choices to construct a traffic monitoring based on WSN, such as the ad hoc self-organized network, the mixed mode of short-range and long-range wireless communication and the hybrid mode of wired/wireless communication [164]. One disadvantage [165] of most conventional vehicle detection methods in a traffic control system is that they can only detect the vehicle in a fixed position. The hybrid mode WSN can be used to detect and monitor the vehicles dynamically that consists of multi layers.

6. Extension Neural Network(ENN) Model: The extension neural network

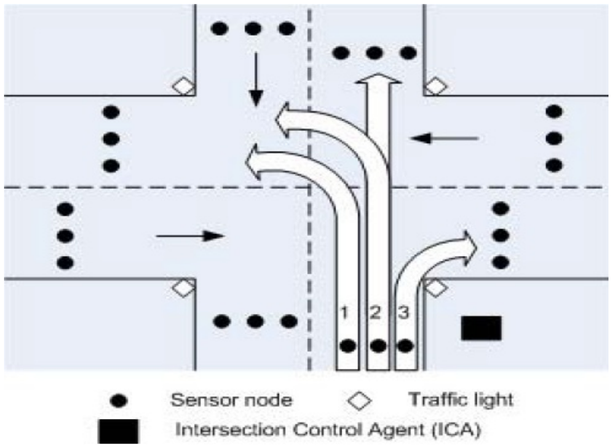


Figure 4.7: Traffic Signal Intersection with Sensors

(ENN) [193]-[195] consists of extension theory and a neural network that uses a modified extension distance (ED) to measure the similarity between data and a cluster center. ENN is another traffic light control system developed to deal with object recognition in outdoor environments. In outdoor environments, lighting conditions cannot be controlled or predicted, objects can be partially occluded, and their position and orientation is not known a priori. The chosen objects are traffic or road signs, due to ease of sign maintenance and inventory in highways and cities, driver support systems and intelligent autonomous vehicles. A genetic algorithm is used for the detection step, allowing localization invariance to changes in position, scale, rotation, weather conditions, partial occlusion, and the presence of other objects of the same color. A neural network can achieve classification.

7. Reinforcement Learning (RL) Models: In this model ([185]-[187]) they use machine learning framework which attempts to approximate an optimal decision-making policy. RL is a field of study in machine learning where an agent, by interacting with and receiving feedback from its environment, attempts to learn an optimal action selection policy. RL algorithms typically learn and progress in an iterative manner. During each iteration, the

agent observes its current environment, from which it infers the environments state, then executes an action that leads the agent to the subsequent state. Next, the agent evaluates this action by the reward or penalty it has incurred and updates a value function, accordingly. The value function is the utility construct that it attempts to maximize (or minimize). A commonly used RL algorithm is Q Learning which is a model-free RL algorithm, it assumes that the agent has no explicit knowledge of its environments behavior prior to interacting with it. Interaction with the environment is what offers the agent knowledge regarding both state transitions (as a function of actions taken) as well as their related long-term reward prospect. The goal of the agent is to maximize such long-term reward, by learning a good policy which is a mapping from perceived states to actions.

8. Algorithm-based Models: the famous algorithm used for traffic light control model is the Genetic algorithm [189] that uses the rules of nature. The great advantage of GAs is the fact that it provides a solution through evolution, but this is also the greatest disadvantage. Evolution is inductive. In nature, life does not necessarily evolve towards a good solution; it can evolve away from bad circumstances. This can potentially cause a species to evolve into an evolutionary dead end.

9. Fuzzy Logic Models: Fuzzy logic [139, 140], [182]-[184] offers a formal way of handling terms like more, less, longer etc., so rules like if there is more traffic from north to south, the lights should stay green longer can be reasoned with. The fuzzy logic controller determines the time that the traffic light should stay in a certain state, before switching to the next state. The order of states is predetermined, but the controller can skip a state if there is no traffic in a certain direction. The amount of arriving and waiting vehicles are

quantized into fuzzy variables, like many, medium and none. The activation of the variables in a certain situation is given by a membership function.

10. Vision-based Models: Video sensors [195] (video and image processing) have become particularly important in traffic applications, mainly due to their fast response and easy installation, operation and maintenance. They also have the ability to monitor wide areas. Intelligent systems may use cameras to extracting useful information such as traffic density and vehicle types (big: truck, middle: van, or small: car) from the camera systems which is very helpful for traffic management specially in the mega cities. Detection of moving objects including vehicle, human, etc. in video can be achieved in different approaches: Temporal difference, optical flow,contour extract and background subtraction. In addition, different classification techniques have been employed after the moving objects are detected in order to identify the moving object (e.g. support vector machines and Neural networks).

For the classification of the traffic control systems, in the literature they are classified into the following based on performance categories:

- Uncoordinated Control: No coordination among traffic signals and provides local intersection control strategies.
- Time-Based Coordinated Control: Provides basic coordination like time of day or day of week. Simple to implement but requires timing plan maintenance.
- Interconnected Control: maintains time plan tables.
- Traffic-Adjusted Control: Critical intersection control (centralized architecture only) and for Local intersection strategies.

- Traffic-Responsive Control: Maintains concept of cycle but changes timing plans more rapidly than traffic adjusted control.

- Traffic-Adaptive Control:Phase change based on prediction from traffic measurement at each signalized approach.

Several real-time traffic signal control systems [196]-[198] for urban networks have been developed in the past few decades. Some of these strategies have been implemented in real-life conditions while others are still in the research and development stage. The authors in [207] classified the strategies into two principal classes of signal control strategies. In the first class, strategies are only applicable to (or efficient for) networks with undersaturated traffic conditions, whereby all queues at the signalized junctions are served during the next green phase. In the second class, the strategies applicable to networks with oversaturated traffic conditions, whereby queues may grow in some links with an imminent risk of spillback and eventually even of gridlock in network cycles.

In the following list, we will highlight the most well-known traffic control systems:

1. Real Time Hierarchical Optimized Distributed Effective System (RHODES): Since 1991, the University of Arizona has been developing a real-time traffic adaptive control system called RHODES that attempts to take advantage of the natural stochastic variations in traffic flow to improve performance. RHODES consists of a three-level hierarchy that decomposes the traffic control problem into three sub problems, network loading, network flow control and intersection control. Algorithms at each level of the hierarchy act upon real-time inputs from the traffic network to make proactive control decisions to reduce delay, improve progression and reduce congestion for travelers. RHODES uses a peer-to-peer communications approach to communicate

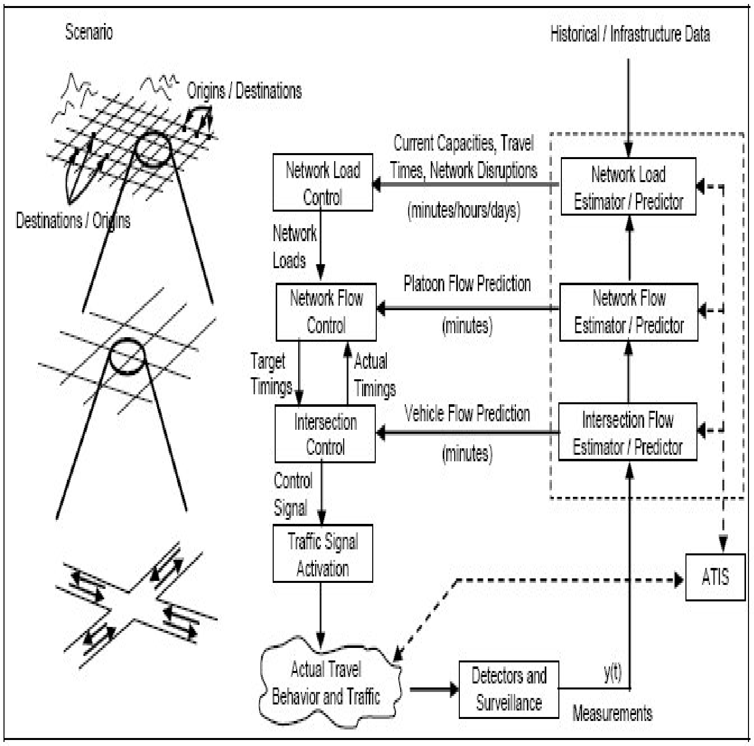


Figure 4.8: Real Time Hierarchical Optimized Distributed Effective System, [197]

traffic volumes from one intersection to another in real-time. By passing the data back and forth over a high-speed communication network, RHODES is able to predict the impacts of traffic arriving 45-60 seconds upstream and plan for traffic phase sequence and phase durations accordingly. RHODES continually re-solves its planned phase timings, every 5 seconds, to adapt to the most recent information. RHODES requires upstream and stop-bar detectors for each approach to the intersections in the network and has a wide variety of parameters that are used to calibrate the traffic model to real-world conditions. RHODES overrides the local controller by sending hold and force-off commands to the controller to set the exact duration of each phase.

2. Optimized Policies for Adaptive Control (OPAC) Virtual Fixed Cycle: The OPAC adaptive control system uses a predictive optimization with a rolling

horizon. This congestion control strategy, which attempts to maximize throughput, adjusts splits, offsets, and cycle length, but maintains the specified phase order. For un-congested networks, OPAC uses a local level of control (at the intersection) to determine the phase durations, and a network level of control for synchronization which is provided either by fixed-time plans (obtained offline), or by a virtual cycle (determined online). The levels of local and global influence are flexible and can be adjusted by the traffic engineer. The state of the system is predicated using detectors located approximately 10-15 seconds upstream on the approaches to the intersection. OPAC sends hold and force off commands to the local controller to set the exact duration of every phase on the signal.

3. Adaptive Control Software Lite (ACS-Lite): ACS-Lite was developed to reduce the costs to deploy adaptive control systems, by consolidating the adaptive processing into a master control unit that supervises local field controllers. ACS-Lite downloads new split, offset, and cycle parameters to the local controllers every 5-15 minutes in response to changing traffic conditions. ACS-Lite is based on a very simple traffic model that has very few tunable parameters and requires modest calibration. Of all actuated systems, ACS-Lite may be the slowest to respond to rapid changes in traffic flows. ACS-Lite sends cycle, offset, and split values to the local controller. The gap-out and force-off logic of the controller works normally with the updated parameters.

4. Split Cycle Offset Optimization Technique (SCOOT): Developed in the United Kingdom, SCOOT is the most widely deployed adaptive system in existence. SCOOT uses both stop-line and advance detectors, typically 150-1,000 feet (50-300 meters) upstream of the stop line or exit loops, loop detec-

tors located downstream of the intersection, measuring vehicles leaving the upstream detector. The advance detectors provide a count of the vehicles approaching each junction. This gives the system a high-resolution picture of traffic flows and a count of the number of vehicles in each queue, several seconds before they touch the stop line (allowing time for communication between the traffic signal controller and the central SCOOT computer). SCOOT also provides queue length detection and estimation. Under the SCOOT system, green waves can be dynamically delayed on a 'just in time' basis based on the arrival of vehicles at the upstream detector, which allows extra time to be allocated to the previous green phase, where warranted by heavy traffic conditions. SCOOT controls the exact green time of every phase on a traffic controller by sending hold and force-off commands to the controller. The SCOOT model utilizes three optimizers: splits, offsets, and cycle. At every junction and for every phase, the split optimizer will make a decision as to whether to make the change earlier, later, or as due, prior to the phase change. The split optimizer implements the decision, which affects the phase change time by only a few seconds to minimize the degree of saturation for the approaches to the intersection. During a predetermined phase in each cycle, and for every junction in the system, the offset optimizer makes a decision to alter, all the offsets by a fixed amount. The offset optimizer uses information stored in cyclic flow profiles and compares the sum of the performance measures on all the adjacent links for the scheduled offset and the possible changed offsets. A SCOOT system is split into cycle time regions that have pre-determined minimum and maximum cycle times. The cycle optimizer can vary the cycle time of each REGION in small intervals in an attempt to ensure that the most heavily loaded NODE in the system is operating at 90% saturation. If all stop bars are operating at less than

90% saturation, then the cycle optimizer will make incremental reductions in cycle time.

5. Sydney Coordinated Adaptive Traffic System (SCATS): Developed in Australia, SCATS uses a split plan selection technique to match traffic patterns to a library of signal timing plans and scales those split plans over a range of cycle times. SCATS gathers data on traffic flows in real-time at each intersection. This data is fed to a central computer via the traffic control signal. The computer makes incremental adjustments to signal timing based on second by second changes in traffic flow at each intersection. SCATS performs a vehicle count at each stop line and measures the gap between vehicles as they pass through each junction. As the gap between vehicles increases, green time efficiency for the approach decreases, and SCATS seeks to reallocate green time to the greatest demand. SCATS selects a timing plan on the controller, and thus the local actuated controller uses its own inherent gap-out and force-off logic to control the intersection second by second.

SCOOT and SCATS [192]-[196] are two well-known and widely-used coordinated traffic-responsive strategies that **function effectively when the traffic conditions in the network are below saturation**, but their performance may deteriorate when severe congestion persists during the peak period. Other elaborated model-based traffic-responsive strategies such as PRODYN and adaptive like RHODES, **employ dynamic programming** while OPAC employs **exhaustive enumeration**. Due to the **exponential complexity of these solution algorithms**, the basic optimization kernel is not real-time feasible for more than one junction.

4.4 Problem Statement

The problem that we are trying to study and discuss in this chapter is related to multiple signalized traffic intersections coordination and control where we need to allow maximum platoon movements with minimum number of stops and make it zero if possible. This will help in achieving several objectives from the list we have mentioned at the beginning of this chapter. Minimizing the waiting time and hence lower trip time as well the queue length. Also, maximizing the service time , cycle time and we can achieve more from the objectives list we have mentioned earlier if we want. The control strategies we have used are all networked based and we considered the network side effects in our design.

4.5 Traffic dynamics and problem definition

Controlling the traffic light intersection requires a prior knowledge of that intersection and the traffic load to be able to set the proper parameters for the control algorithm, especially if the system used is not an intelligent system like time based traffic control. Basically most of the traffic signals intersections have four directions queues, North (N), South (S), East (E) and West (W) as shown in Fig. 4.9. The other possible queues are North West (NW), South East (SE), East South (ES) and West North (WN) as shown in the Fig. 4.10. The model in Fig. 4.10 simply shows that two directions can be open at the same time, for example, N and S direction will move then W and E at the same time because there is no turning in other directions (also it is called two phases intersections) like NW or SE. The other scenario is when we have the other directions NW, SE, EN and WS, (we call it four phases intersection) then the control algorithm will be more complicated and more sensing elements are required. For simplicity, we will give a number for each queue q_i where $i = 1, ..., 8$ in the following order

$(N, S, E, W, NW, SE, EN, WS)$

The intersection consists of four streets with 8 possible queues , assuming all right side movements are free and do not require a signal. The state equation for the continuous traffic flow process associated with any movement i that is sampled every Δt seconds, where time is indexed with the integer k, can be expressed by the current queue $q_i(k)$:

$$\begin{aligned} q_i(k+1) &= q_i(k) + \Delta q_i(k) + \Delta p_i(k), i = 1, 2, ...8 \\ \Delta q_i(k) &= q_i^{in}(k) - q_i^{out}(k) \\ \Delta p_i(k) &= p_i^{in} - p_i^{out} \end{aligned} \tag{4.1}$$

where $q_i^{in}(k)$ is the number of incoming new vehicles at time interval $[k-1, k]$ in link or queue i, $q_i^{out}(k)$ is the number of vehicles able to pass the intersection during the green signal interval Tg from link or queue i, also Tg can be called as the control interval , $q_i(k-1)$ is the queue of vehicles waiting for the green signal to happen at time k, $\Delta p_i(k)$ represents the fluctuation between a parking lot and link i or the effects of any non-controlled intersection between any two intersections where p_i^{in} is used for vehicles have left the parking or came from non-controlled intersection and joined the traffic in the queue i and p_i^{out} is used for vehicles which left the queue i and went for a parking or went into a sub road or what we call non-controlled intersection. These disturbing flows (see Fig. 4.11) can be considered either as disturbance or as known perturbations if they can be well measured or estimated. In case these uncertainties or perturbations are unknown and can't be measured, then robust control system is needed.

The output $q_{out}(k)$ can further be expressed as a function of the current control of the intersection, $u(k)$, and the current queue, $q(k)$:

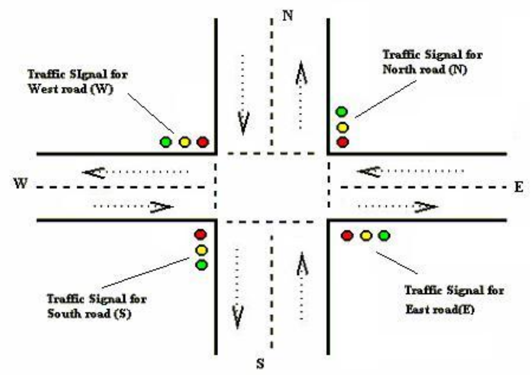


Figure 4.9: Basic Traffic Signal Intersection Control

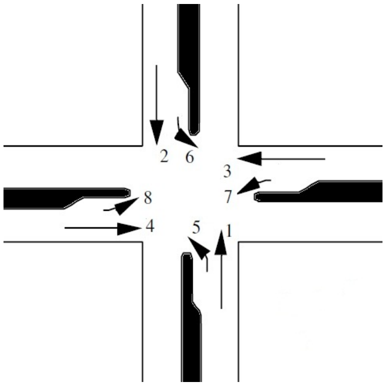


Figure 4.10: Typical Traffic Signal Intersection Control

$$q_{out}(k) = f_{out}(u(k), q(k)) \tag{4.2}$$

The general discrete LTI state space representation is the following:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Fd(k) \\ y(k) &= Cx(k) \end{aligned} \tag{4.3}$$

Using equation 4.3, it is possible to describe the dynamics of a traffic network with the following: The state matrix A is considered as an identity matrix. The elements of the state vector $x(k)$ represent the number of vehicles of each con-

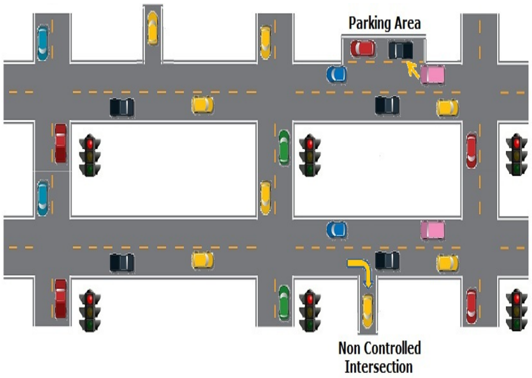


Figure 4.11: Queues uncertainties in traffic between intersections

trolled link or in other words the queue length in that lane and the number of states is equal to the number of controlled links in the network. The second term of the state equation is the product of input matrix B and control input u where the vector u contains the green times of all stages. Matrix B can be constructed by the appropriate allocation of the combinations of saturation and turning rates. Their numerical values are the results of a corresponding controller at each cycle. The diagonal values of B are negative and represents the saturation flow and the product of $B_{ij}u_i$ where $i = j$, diagonal elements shows the outflow from link i. The other elements in B_{ij} where $i \neq j$ contains the turning rates from link i to link j. Naturally the number of states is equal to the number of controlled links in the network. The product $Bu(k)$ arises from difference of in and out flow for the traffic in the link or queue i during the control interval. Each output inside of the network is a measured state (number of vehicles of the link i) that makes the output equation simplified to $y(k) = x(k)$ and $C = I$. Finally, the traffic coming from non-controlled intersections or parking are considered as disturbance to the system in $d(k)$. The eq. 4.3 can be rewritten as :

$$\begin{aligned} Q(k+1) &= AQ(k) + BG(k) + Fd(k) \\ Q^{out}(k) &= CQ(k) \end{aligned} \tag{4.4}$$

where $Q(k)$ is a vector of queues information for all the eight directions showing in Fig. 4.10, and $G(k)$ contains the green timing for each direction.

$$\begin{aligned} Q(k) &= [q_1(k)\ q_2(k)\ .....\ q_8(k)]_j^t \\ G(k) &= [Tg_1(k)\ Tg_2(k)\ .....\ Tg_8(k)]_j^t \end{aligned} \tag{4.5}$$

Following the same manner, we can generalize that to traffic networks with multiple intersections. In a traffic network with n intersections, the order of the dynamic equations is increased to $n \times m$ where m is the number of possible movements in that intersection, for example, in Fig. 4.14 we have $m = 8$ for any intersection. However, any complicated traffic network can be decomposed into a group of small ”elementary networks”, with similar intersections. In this manner, the study of the entire traffic network can be reduced to the analysis of these elementary networks and the inter-connections.

Flow characteristics of traffic are fundamental in analyzing intersection delay or capacity. Vehicles occupy space and, for safety, require space between them. With vehicles moving continuously in a single lane, the number of vehicles passing a given point over time will depend on the average headway or the average arrival rate per unit time. Two factors influence capacity at a signalized intersection:

- Conflicts occur when two vehicles attempt to occupy the same space at the same time. This requires allocation of right-of-way to one line of vehicles while the other line waits.

- The interruption of flow for the assignment of right-of-way introduces additional delay. Vehicles slow down to stop and are also delayed when again permitted to proceed.

These factors (interruption of flow, stopping, and starting delay) reduce capacity and increase delay at a signalized intersection as compared to free-flow operations. Vehicles that arrive during a red interval must stop and wait for a green indication and then start and proceed through the intersection. The delay as vehicles start moving is followed by a period of relatively constant flow.

The green signal period given for each side or combination of directions will be called **phase** (see Fig. 4.12). The combination of phases can be called as **Cycle** where each phase or cycle must not exceed certain period to maintain the fairness for all directions in that intersection and it shall not be less than certain minimum. In all situations, the phases time shall not push the situation in that intersection to exceed the saturation level which will lead to traffic jam as we can see from Fig. 4.22. Phasing reduces conflicts between traffic movements at signalized intersections. A phase may involve:

- One or more vehicular movements.
- A combination of vehicular and pedestrian movements.
- One or more pedestrian crossing movements.

The National Electrical Manufacturers Association (NEMA) has adopted and published precise nomenclature for defining the various signal phases to eliminate misunderstanding between manufacturers and purchasers. Fig. 4.13 illustrates a 4-phase sequence separating all vehicular conflicts. Holding the number of phases to a minimum generally improves operations. As the number of phases increases,

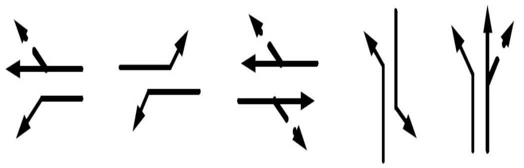


Figure 4.12: Different Phases for Traffic Signal Intersection

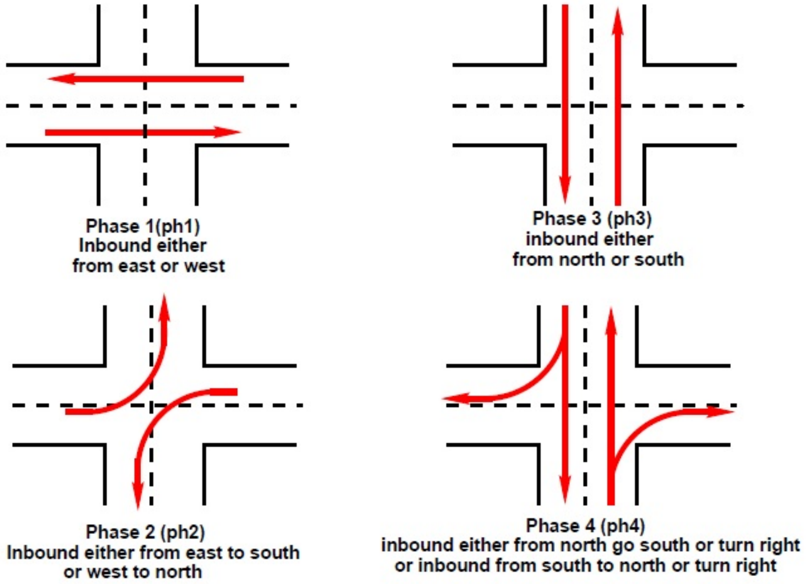


Figure 4.13: Example of 4-Phase intersection, [173]

cycle lengths and delays generally increase to provide sufficient green time to each phase. The goals of improving safety (by adding left-turn phases) and operations at a signalized intersection may conflict, particularly with pre-timed control.

Full-actuated traffic control illustrates variable-sequence phasing. In Fig. 4.11, all approach lanes have detectors, using these detectors; actuated control skips phases with no traffic present and terminates certain movements when their traffic moves into the intersection. This capability produces a variation in the phasing sequence. The phasing options selected may be changed with the signal timing plan.

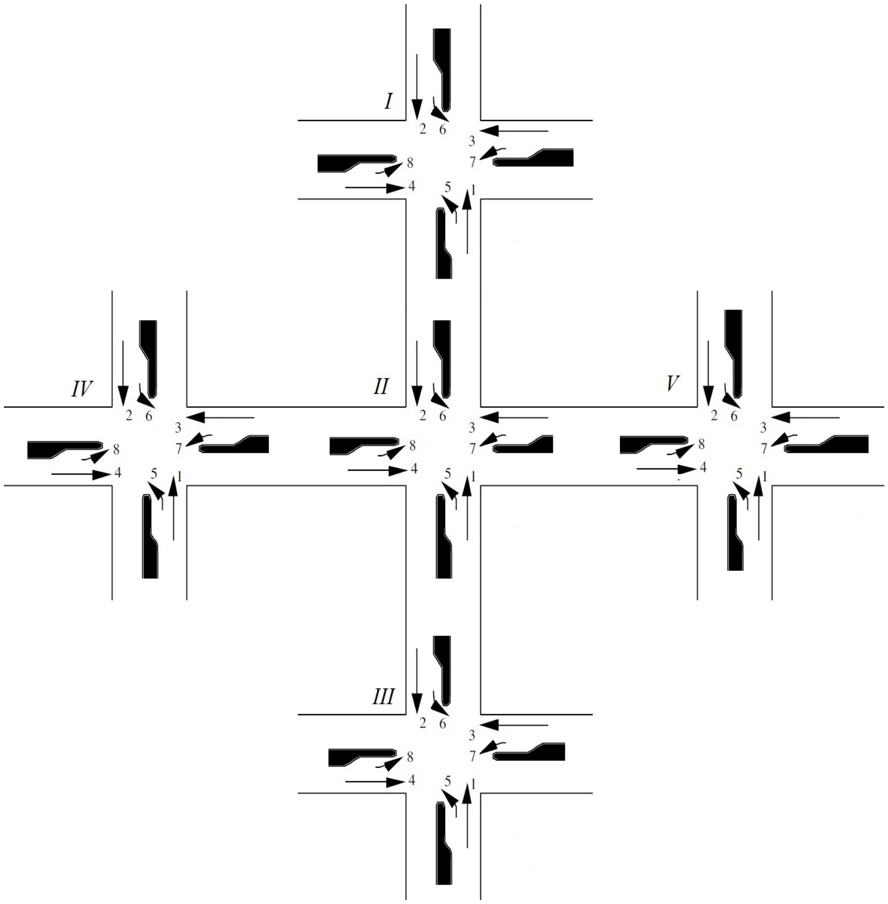


Figure 4.14: A Traffic network with five intersections

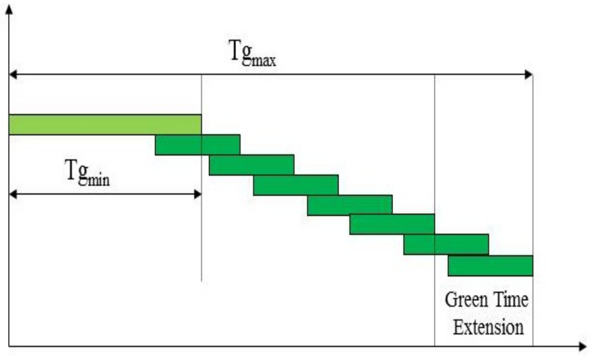


Figure 4.15: Green Time Extension

4.5.1 Constraints of traffic signal control

There are several constraints which have to be taken into account and these can be determined by the geometry of the traffic network and we can list them as follows:

- Queue/Link Capacity: is defined by the maximum number of vehicles for link i and it can be determined by the length of link between two intersections. so $0 \leq q_i(k) \leq q_{i,max}(k)$.
- Control Constraint: the maximum Tg is the time interval (in seconds) of green time for link or queue i and shall not exceed a certain value to be fair to other links during the same cycle time. $Tg_{i,min} \leq Tg_i(k) \leq Tg_{i,max}$, see Fig. 4.15.
- Waiting Time: the time Tw spent by vehicles waiting until the signal becomes green. It is very important to **minimize** this time as much as possible by providing good service mechanism at the signalized intersection. This parameter can be calculated for direction i at intersection j by taking the sum of all other directions green time, or simply the phases because two directions can be in one phase so it is easier to use the phase p, in the same

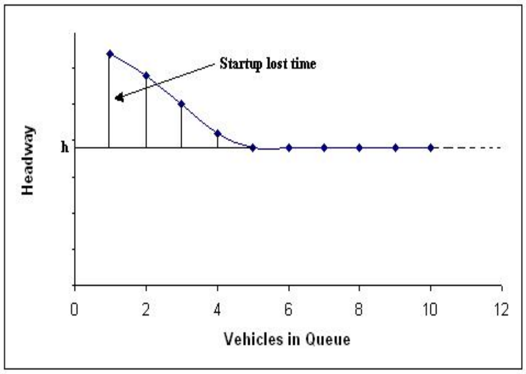


Figure 4.16: Startup Delay

intersection j.

$$\begin{aligned} Tw_{ij} &\geq \sum_{m \neq i} [Tg_{mj}] \\ m &= 1, 2, 3, ...p \end{aligned} \tag{4.6}$$

As an example, if we have four phases, the estimated Tw_1 which is the same as Tw_2 because both are in phase $p1$, will be:

$$Tw_{p1} = Tg_{p2} + Tg_{p3} + Tg_{p4}. \tag{4.7}$$

we can see a direct relation between Eq. 4.4 and Tw. Theoretically, The total service time Ts required for one phase to pass all the cars waiting in a queue q_i is dependent on τ_s the service time required to pass one row of cars at the same time and the physical structure of the street. Here we mainly focus on the number of lanes. So, $T_s = (\tau_s * q_i)/No.Of.Lanes$.

- Startup Delay: as part of Tg there is an important component which is the startup delay time T_d where the drivers take few seconds sometimes to realize the green LED is ON. Signal indication turns from red to green and vehicles do not instantly move at the saturation flow rate.

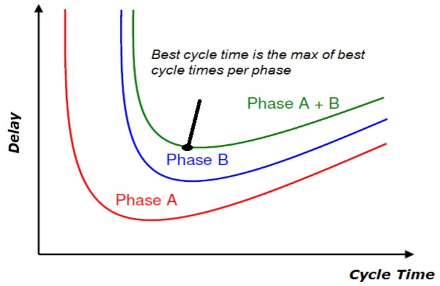


Figure 4.17: Best cycle time

- Cycle Time: the cycle time Tc is the time to complete the execution of all phases for the intersection and it shall be bounded by a certain value, $Tc_j \leq Tc_{max}$. It is also possible to choose the maximum best cycle time of any phase in that cycle as shown in Fig. 4.17. The cycle time may vary due to traffic situation. In case of heavy traffic, the best way is to have long cycle times to maximize steady-state flow. In contrast, when the traffic is light the better is the short cycle time to minimize the delays for vehicles. Another important issue for the cycle time selection is the one related to the nature of intersection control whether it is for single or multiple intersections. For single Intersection, the ratio of $R_i(k) = (flow_{in}/flowout_{max}) < 1$ shall not reach $R_i(k) = 1$ which is the saturation level and the traffic jam will occur at $R_i(k) > 1$ which is the worst scenario. If this happen, then phase time shall be recalculated to have the ratio $R_i(k)/\sum_i(R_i(k))$. In case of Multi-intersections, we need to have careful timing to achieve the best throughput with good platoon management to make the flow of vehicles smooth through several intersections with less delay (green-wave progression or successive green signals) and to minimize overall delay and/or number of stops. This concept is explained in Fig. 4.18 where we can see the platoon of vehicles are moving in the two parallel directions in manner that number of stops are minimized and this because of proper coordination between intersection

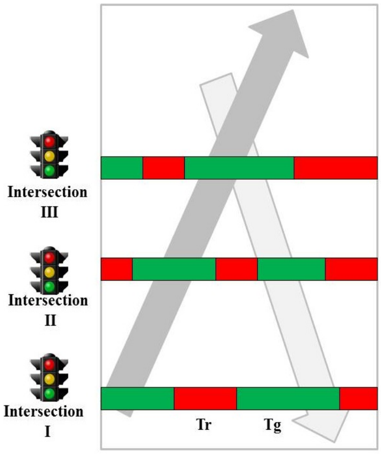


Figure 4.18: Multi-intersection Control Timing with Coordination

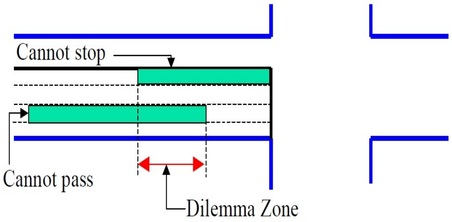


Figure 4.19: Dilemma Zone

controllers.

- Phase: a phase is any period in a cycle where non-conflicting traffic movements may run. It is very important the selection of phase type and how many phases are required for each intersection.
- Dilemma Zone: a dilemma zone [199] is a range, in which a vehicle approaching the intersection during the yellow phase can neither safely clear the intersection, nor stop comfortably at the stop-line and it is one of the main contributors to signal-related accidents. Note that both the length and the location of a dilemma zone may vary with the speed of the approaching vehicles, driver reaction times, and vehicle acceleration/deceleration rates (this will not be considered in this work).
- Lost Time T_L: It is the non-utilized time in case the Tg given is more than

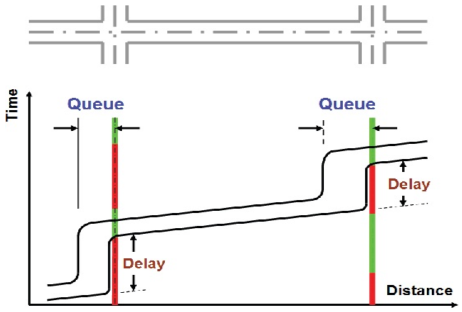


Figure 4.20: Traffic Intersections without Coordination

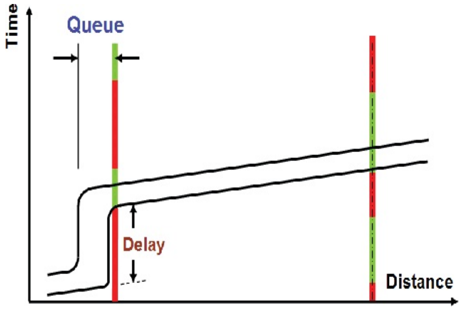


Figure 4.21: Traffic Intersections with Coordination

required, there will be time period not used by any car while others are waiting.

- Safe Time: it is the time required as gap time (Red Clearance interval) between the current signal which has just turned red for the current phase and the start of green time for the next phase. This gap or safe time T_s is required for safety to avoid or minimize the crashes between cars crossing the red signal at the last moment while ending the current running phase.
- Number of Stops: another purpose of coordination is to minimize the overall delay and/or number of stops. This can be achieved using fixed-timing plans or using adaptive technology. The three main components of coordinated timings are: (1) Cycle time (2) Stage splits - the amount of time allocated to a phase in a cycle (3) Offsets - green signals at adjacent intersections are set to occur at a given time, relative to that at a reference intersection.

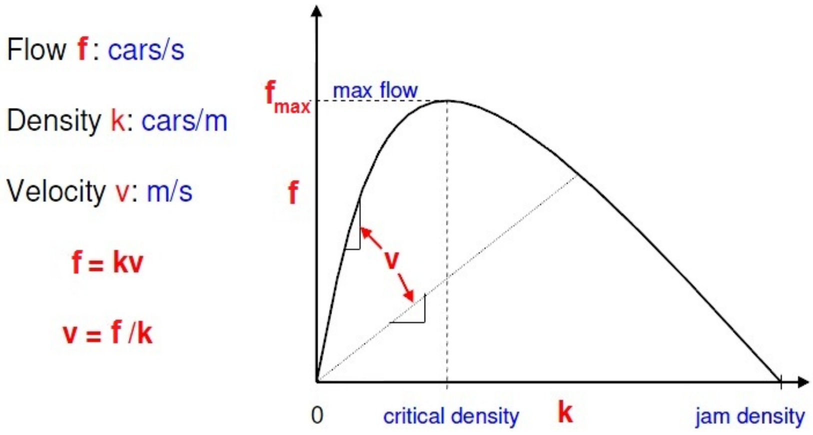


Figure 4.22: Traffic Flow Density Relation

4.6 Communication link impacts

The use of communication link between sensors and the intersection controller will introduce some network issues due to the nature of this shared link and we have already discussed in details this in Chapter 3 . The intersection controller will run based on the pre-timed tables in case of any significant delay or many packet dropouts to avoid open-loop problem which will lead to huge traffic accumulation and violations. That will continue for one cycle until the next data arrives properly, otherwise after a certain number of similar problems, alarms will be sent to the traffic control room operator for maintenance and troubleshooting. For example, suppose that sensors packets for queue arrival are dropped, then the intersection controller will work on the last value received. If the problem is still not resolved in the next cycles, the controller will run based on a default time for Tg regardless of the queue length. Another solution, is to work based on a table that contains the historical data averages for similar day time, e.g. the peak hours will be different from normal hours. More details on these impacts will be presented and explained in the simulation section.

4.7 Decentralized Networked Control Structure

The computational complexity of a large traffic network can be reduced efficiently by dividing the network into small intersections, and controlling the local intersection controllers separately in a decentralized structure over a communication network. The traffic flow interactions between intersections are cut off (or disconnected) [85], and will be considered constant and known by each intersection in advance. Because the estimates of the input traffic flows from other intersections may be far from the real values, the local controllers may not be able to find the real optimal solutions for the intersections. Moreover, since the intersection are completely disconnected, the overall performance of the whole network will be deteriorated when we have a high traffic flow between intersections along that highway.

By applying this structure we will have the generalized model for the system shown in Fig. 4.14 that has 5 traffic light intersections as the following:

$$\begin{aligned} Q(k) &= [Q_1 \; Q_2 \; .... \; Q_j], j = 1, 2, ...5 \\ Q_j(k) &= [q_{1,j}(k) \; q_{2,j}(k) \; .... \; q_{8,j}(k)]^t, \\ q_{i,j}(k) &= q_{i,j}(k-1) + \Delta q_{i,j}(k), i = 1, 2, ...8, \\ \Delta q_{i,j}(k) &= -q_{i,j}^{out}(k) \end{aligned} \tag{4.8}$$

Here $\Delta q_{i,j}(k)$ is negative (-) because we don't consider the incoming traffic from other intersections since we assume there is no interactions between the intersections, and hence the state space model will be

$$\begin{aligned} x_j(k+1) &= A_j x_j(k) + B_j u_j(k) \\ y_j(k) &= C_j x_j(k) \end{aligned} \tag{4.9}$$

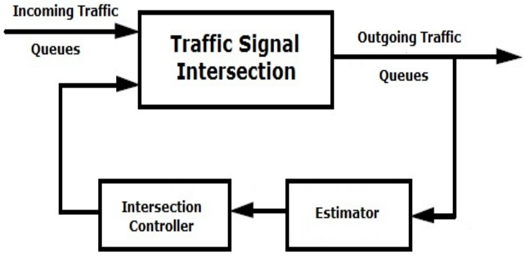


Figure 4.23: Traffic controller using an estimator at the intersection.

where j represents the intersection number, as we can see from this structure that the queues information between intersections are not known in advance so we may use an estimator to help the intersection controller to perform better by having some estimates about the new queue length considering the outgoing traffic as shown in Fig. 4.23.

The assumption in this structure is a full decentralization where all intersections are fully isolated and each one is working independently from each other which is the case in many intersections in several countries. So, we will not discuss the communication link effects for this structure between intersections controllers. However, the data collection for each lane coming to an intersection is communicating over the shared link with the intersection controller which will address some of the issues mentioned earlier.

4.8 Distributed Networked Control Structure

DNCS uses local controllers for different subsystems where the local controllers exchange information and coordinate between each other. Therefore, each local controller will make its own decisions based on both information from the subsystem itself and the information obtained from other subsystems. The more information the local controllers have, the better overall performance and stability of the whole traffic network will be achieved. However, if the amount of information that the local controllers take into consideration of increases, the computational complexity will become very high and this will affect the stability because of a high computation time that may delay the response to the traffic situation.

By applying this structure we will have the generalized model for system shown in Fig. 4.14 that has 5 traffic light intersections as the following:

$$\begin{aligned}
Q(k) &= [Q_1\ Q_2\ ....\ Q_j], j = 1, 2, ...5 \\
Q_j(k) &= [q_{1,j}(k)\ q_{2,j}(k)\ ....\ q_{8,j}(k)]^t, \\
q_{i,j}(k) &= q_{i,j}(k-1) + \Delta q_{i,j}(k), i = 1, 2, ...8, \\
\Delta q_{i,j}(k) &= q^{in}_{i,j}(k) - q^{out}_{i,j}(k)
\end{aligned} \tag{4.10}$$

as we can see that we consider all the incoming traffic from other intersections where for example in Fig. 4.14 the traffic coming from intersection III from queue $6, 3$ will affect the queue in intersection I in queue $6, 8$ and so on. and hence the state space model will be

$$\begin{aligned}
x_j(k+1) &= A_j x_j(k) + B_j u_j(k) + H_j(k), \\
y^j_k &= C_j x_j(k) + W_j(k) \\
u^j_k &= K_j x_j(k) + M_j(k)
\end{aligned} \tag{4.11}$$

where $H_j(k) = \sum_{n=1,n\neq j}^{5} A_{n,j} x_n(k)$ that contains the information about the other intersections queues that may help the current intersection in case of long queue there to pro act to minimize the vehicles accumulation in that lane, $W_j(k) = \sum_{n=1,n\neq j}^{N} C_{n,j} x_j(k)$ to show the information about the output queues from other intersections that is exchanged between the controllers and

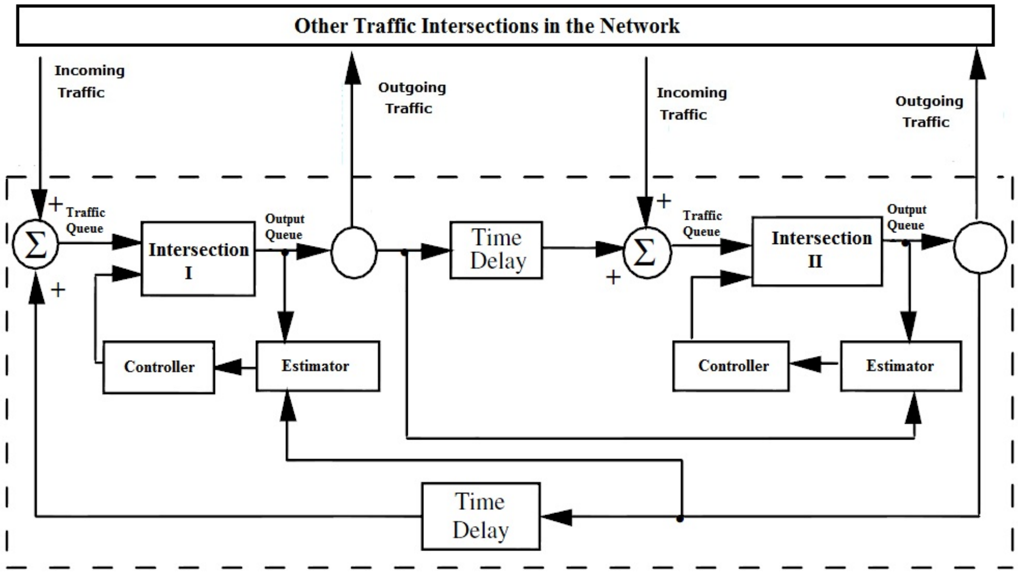


Figure 4.24: Two Intersections Traffic Controllers (DNCS).

$M_j(k) = \sum_{n=1,n\neq j}^{N} K_{n,j}x_j(k)$ that shows the control signals or duration in other intersections. For example, the intersection I controller in Fig. 4.14 will be able to know the status of the signal at lane 6 from intersection III if it is green and also the queue length and the output queue during the green period will be also sent before that to the controller at intersection I, then there could be several scenarios to minimize the queue length at lane 6 in the intersection I by extending the Tg where $Tg < Tg_{max}$, (see Fig. 4.15), if it is green, or give the priority to this side if the other sides in the intersection I has lower queue length, or minimize the the Tg for the other sides if the queue lengths are smaller.

Since we are using a control over a communication network, then we may have some problems due to the use of the shared communication link such as delay, packet dropout, varying sample interval and transmission constraints. From Fig. 4.9 we can see that each intersection will have an information about the other intersection's outgoing queue which will help to get a better estimation and control for the value of Tg and also in case the next traffic signal is too crowded, the preceding intersection controller will try to delay the traffic by using the minimum

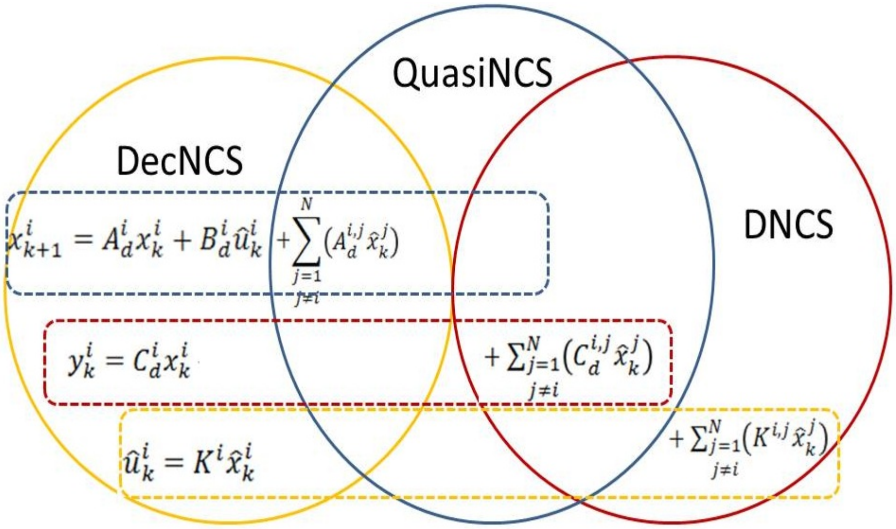


Figure 4.25: QuasiNCS

Tg to avoid sending more traffic to that crowded intersection and hopefully the jam will be released during the next cycle.

4.9 Quasi-Decentralized Networked Control Structure

To solve the problem where a DecNCS structure cannot provide the required stability and performance properties, and to avoid the complexity and high exchange of information required between controller in DNCS , a quasi-decentralized networked control strategy, for simplicity we will call it as **QuasiNCS**, (it is partially decentralized and not fully distributed), (see Fig. 4.25), with minimum cross communication between the intersections offers a suitable compromise and it provides a way of ensuring partial knowledge of how the local controller is affecting the global system and can guarantee certain stability for the overall traffic network.

The term quasi-decentralized networked control refers to a situation in which most signals used for control are collected and processed locally, although some signals (the total number of which is kept to a minimum) still need to be transferred between local units and controllers to adequately account for the interactions between the different units and minimize the propagation of disturbances and process upsets from one unit to another.

$$\begin{aligned} Q(k) &= [Q_1 \; Q_2 \; .... \; Q_j], \\ Q_j(k) &= [q_{1,j}(k) \; q_{2,j}(k) \; .... \; q_{8,j}(k)]^t, \\ q_{i,j}(k) &= q_{i,j}(k-1) + \Delta q_{i,j}(k), i = 1, 2, ...8, \\ \Delta q_{i,j}(k) &= q_{i,j}^{in}(k) - q_{i,j}^{out}(k) \end{aligned} \tag{4.12}$$

where $q_{i,j}^{in}(k)$ is the incoming new vehicles at time interval $[k-1, k]$ for intersection j for queue lane number i , $q_{i,j}^{out}(k)$ is the number of vehicles that were able to pass the intersection j during the green signal interval, Tg for the queue lane i at that intersection and $q_{i,j}(k-1)$ is the queue of vehicles that were waiting for green signal to happen at time k.

The discrete state space for the generalized model with multiple intersections can be shown to be as follows:

$$\begin{aligned} x_j(k+1) &= A_j x_j(k) + B_j u_j(k) + H_j(k), \\ y_k^j &= C_j x_j(k) \end{aligned} \tag{4.13}$$

where $H_j(k)$ already defined in eq. 4.11.

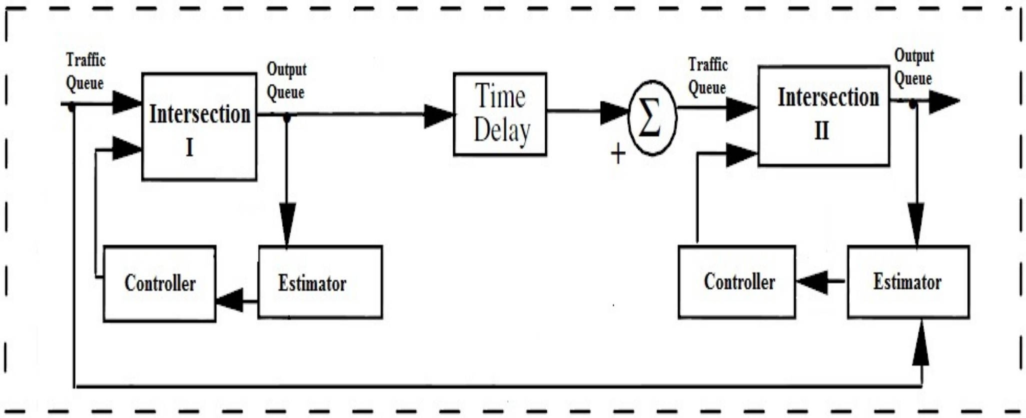


Figure 4.26: Two Intersections Traffic Controllers (QuasiNCS).

4.10 Hierarchical Structure

The main aim of this structure is to perform traffic management at a strategic level in urban, interurban or mixed areas. The city or traffic network where the traffic has to be supervised is divided into several sections called problem areas or zones. The decomposition of the city into zones allows for a better analysis and understanding of the causes and evolution of traffic problems than if performed from a global perspective. This split does not define a set of disjointed areas whose sum is the whole city, but every area represents a part of the city where a determined traffic behavior is usually present and where a set of signal elements can be managed to influence this behavior. Then, the zone may overlap with surrounding zones sharing, for instance, some signals but using them from different points of view. So, a problem area or zone is a part of a city where traffic behavior is locally studied and suitable control actions may be defined to improve the traffic state.

Every zone is controlled by a controller, called control agent, which understands the traffic conflicts that may appear, the usual behavior of vehicles in the area and the signal and/or VMS (Variable Message System) actions that may

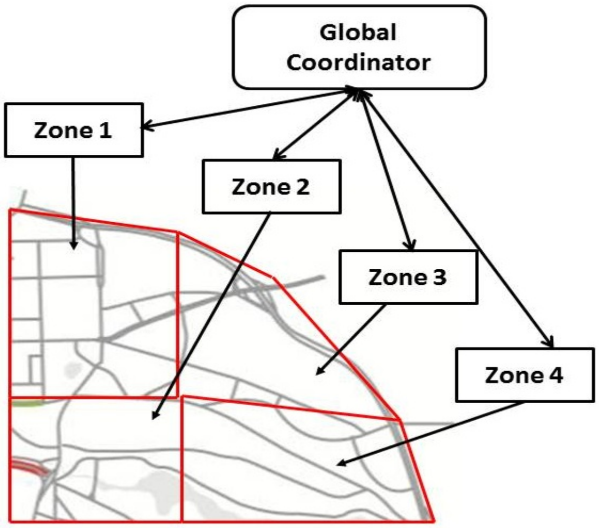


Figure 4.27: Hierarchal Traffic Control - Dividing the Network into Problem Areas

improve the traffic state, supervise every problem area. The control proposals generated by every agent are received by a higher level agent, called the coordinator, whose aim is to produce global proposals for the whole city by putting together the local proposals provided by the agents and removing the inconsistencies among them.

4.11 Traffic Control Closed-loop Models

The most common and systematic approach is to use a dynamic output feedback, where the controller (or compensator) has its own dynamics. The simplest form is an observer structure

$$\begin{aligned} \tilde{x}_j(k+1) &= A_i\tilde{x}_j + B_j u_j + L_j(y_j - C_j\tilde{x}_j) \\ u_j &= -K_j\tilde{x}_j, \quad j = 1, \ ..., \ 5 \end{aligned} \tag{4.14}$$

In this simple approach, $\tilde{x}_j$ is an estimate for the actual x for each subsystem

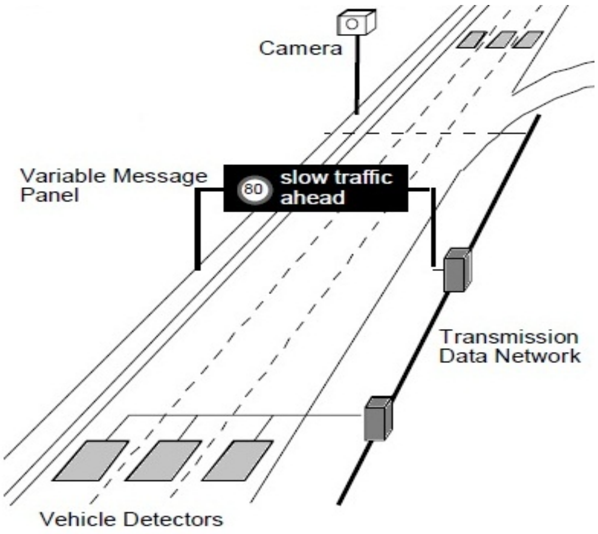


Figure 4.28: Variable Message System

i and we need to pick a good observation gain L_i such that $\tilde{x}_j \to x$ as fast as possible. In this work, we will use observer-based controllers in the sense that for each intersection of the traffic network we have one observer-based controller and the controllers either exchange information or are not based on the selection of the structure from the three we have mentioned previously (DecNCS, QuasiNCS, DNCS) in this work. The i^{th} networked observer-based controller is given by considering the network side effects we have discussed in this work:

$$\begin{aligned} \tilde{x}_j(k+1) &= A_j\tilde{x}_j(k) + B_j\hat{u}_j(k) + O_j + H_j \\ O_j &= L_j\Gamma_j^y(\hat{y}_j(k) - C_j\tilde{x}_j(k)) \\ H_j &= \sum_{j=1,j\neq i}^{N} A_{i,j}\hat{x}_j(k) \\ \hat{u}_j(k) &= -\,K_j\,\tilde{x}_j(k) \end{aligned} \tag{4.15}$$

where $\tilde{x}_j(k+1)$ represents the state estimate at time $(k+1)$ for the plant state $x_j(k+1)$, $B_j = [\int_0^{h-\tau_{rt}^k} e^{As}ds]B$ when $\tau_{rt}^k \leq h$ where h is the sampling interval. The output related matrices $L_j(k),\ K_j,\ j = 1,\ ...,\ 5$ are the subsystem gain matrices. The state estimation error is $\psi_j(k) = \tilde{x}_j(k) - x_j(k)$.

To deal with the communication constraints, the observer structure is used where the standard output is applied only when a new measurement is received. The dynamics of all controllers can be shown in a discrete model that composed of block diagonal matrices written as follows for the DNCS, DecNCS and QuasiNCS

$$\xi_{k,DNCS} := \begin{bmatrix} x_k & \psi_k & e_k^y & H_k & W_k \end{bmatrix}^t \tag{4.16}$$

$$\xi_{k,DecNCS} := \begin{bmatrix} x_k & \psi_k & e_k^y \end{bmatrix}^t \tag{4.17}$$

$$\xi_{k,Quasi} := \begin{bmatrix} x_k & \psi_k & e_k^y & H_k \end{bmatrix}^t \tag{4.18}$$

by combining the foregoing relations, the overall closed-loop dynamics can be expressed as follows for the three control strategies (DNCS, DecNCS and QuasiNCS).

$$\xi_{k+1,DNCS} = \mathcal{A}_{k+1,DNCS}\, \xi_k \tag{4.19}$$

$$\mathcal{A}_{cl,DNCS} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix} \tag{4.20}$$

$$\tag{4.21}$$

$$\begin{aligned} a_{11} &= A_j + B_j K_j, \;\; a_{12} = -A_j + L_j C_j, \\ a_{22} &= A_j - L_j C_j = -a_{12}, \\ a_{31} &= C_j(-A_j + \alpha_k K_j B_j + I), \\ a_{32} &= \alpha_k K_j B_j C_j, \\ a_{33} &= (I - \beta_k \Gamma_j^y), \\ a_{44} &= a_{55} = I, \end{aligned} \tag{4.22}$$

The others non mentioned elements are all zeros.

For the fully decentralized structure in which case the exchange of state information among subsystems is not allowed, then it will be reduced to:

$$\begin{aligned} \zeta_{k,DecNCS} &:= \begin{bmatrix} x_k & \psi_k & e_k^y \end{bmatrix} \\ \zeta_{k+1,DecNCS} &= \mathcal{A}_{DecNCS}\, \zeta_k \\ \mathcal{A}_{DecNCS} &= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & 0 & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \end{aligned} \tag{4.23}$$

and finally, for the QuasiNCS the $\mathcal{A}_{QuasiNCS}$ will be as the following:

$$\begin{aligned} \xi_{k,QuasiNCS} &:= \begin{bmatrix} x_k & \psi_k & e_k^y & H_k \end{bmatrix}^t \\ \zeta_{k+1,QuasiNCS} &= \mathcal{A}_{QuasiNCS}\, \zeta_k \\ \mathcal{A}_{QuasiNCS} &= \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ 0 & 0 & 0 & a_{44} \end{bmatrix} \end{aligned} \tag{4.24}$$

To sum up, the foregoing control structures can be cast into the following generic form

$$\begin{aligned} \zeta_{k+1} &= \mathcal{A}\, \zeta_k \\ \mathcal{A}_{cl} &= blockdiag\{\mathcal{A}_{1,cl},\ ...,\ \mathcal{A}_{N,cl}\} \end{aligned} \tag{4.25}$$

4.12 Stability Analysis

In the sequel, we define a global Lyapunov functional by

$$V = \xi_k^t \mathcal{P} \xi_k,\ \ \mathcal{P} = blockdiag\{\mathcal{P}_1,\ ...,\ \mathcal{P}_N\},\ \mathcal{P}_j > 0 \tag{4.26}$$

Evaluating the first difference ΔV along the solutions of (4.25) yields

$$\Delta V = -\mathcal{P} + \mathcal{A}_{cl}^t \mathcal{P} \mathcal{A}_{cl} \tag{4.27}$$

According to Laypunov stability theorem, a necessary and sufficient condition for stability is $V > 0,\ \Delta V < 0$. The following is a preliminary result

Lemma 4.1 *Given the gains* K *and* L, *system (4.25) is said to be asymptotically stable if there exists symmetric positive definite matrices* $0 < \mathcal{P}_j = \mathcal{P}_j^t \in \Re^{n_i \times n_i},\ 0 < \mathcal{X}_j \in \Re^{n_i \times n_i},\ 0 < \mathcal{Z}_j = \mathcal{Z}_j^t \in \Re^{n_i \times n_i},\ \ i = 1,\ ...,\ NoOfDirections$ *such that the following LMIs*

$$\begin{bmatrix} -\mathcal{P}_j & \mathcal{A}_{j,cl}^t \mathcal{X}_j \\ \bullet & -\mathcal{X}_j - \mathcal{X}_j^t + \mathcal{Z}_j \end{bmatrix} < 0, \quad j = 1,\ ...,\ N \tag{4.28}$$

have a feasible solution for $j = 1,\ ...,\ N$, *where* N *is the number of intersections.*

Remark 4.1 *By looking at the closed-loop matrix (4.20)-(4.11) in the distributed-control case, it is instructive to let the matrix* X *have the following form where the size will be matching the size of* A_{cl} *according to the control structure that we have selected:*

$$\mathcal{X} = \begin{bmatrix} X_{11} & X_{12} & X_{13} & X_{14} & X_{15} \\ 0 & X_{21} & X_{23} & X_{24} & X_{25} \\ 0 & 0 & X_{33} & X_{34} & X_{35} \\ 0 & 0 & 0 & X_{44} & X_{45} \\ 0 & 0 & 0 & 0 & X_{55} \end{bmatrix} \tag{4.29}$$

Indeed, the decentralized and quasi-decentralized cases can be treated in a similar way.

Proof. That $\mathcal{P}_j > 0$ implies that $V > 0$. Applying **Lemma** 4.1 to inequality $\Delta V < 0$ using (4.27) with $\mathcal{M} = \mathcal{P}_j$, $\mathcal{N} = \mathcal{A}^t_{j,cl}$ and invoking Schur complements, we readily obtain inequality (4.28). ▮

4.13 Uncertainties and Robust Control

In the previous sections, the general LTI state space representation of the urban traffic system was shown and discussed with several details. The possible state uncertainties were neglected or considered as known parameters, the demand and exit flows are known values within the link. Typically, state uncertainties appear due to unexpected traffic fluctuations caused by parking places along the road or non-controlled junctions in the network (Fig. 4.6). The measurements of these disturbing flows would lead to enormous costs in urban network. Therefore, it is more reasonable to treat them as bounded uncertainties. A common approach for modeling uncertainties is the use of bounded additive disturbance model. Another potential technique is the multiplicative approach which may involve state uncertainties in the traffic model. However, this section will be for future work extensions and we will not discuss it in the simulation.

$$\begin{aligned}
\Delta x_i(k) &= x_i(k) - x_i^N(k), i = 1, 2, ...8 \\
\Delta u_i(k) &= u_i(k) - u_i^N(k) \\
\Delta d_i(k) &= d_i(k) - d_i^N(k) \qquad (4.30)
\end{aligned}$$

4.14 Simulation Studies

In the simulation we considered 5 intersections and we tried to compare the results from the proposed approaches. The simulation was done using MATLAB 2008 on Laptop with Windows 7 Professional, 2.73 GHz with 8 cores and 8 GB Memory. The following assumptions were used :

- Distance between each intersection is known (1 km).
- Average speed is 80 Km/h.
- Each road has 3 main lanes.
- One service lane for left direction and one for right.
- Flow of traffic is smooth and no major interruption.
- Communication between sensors to controller is over a lossy network.
- Cross-communication between each intersection controller is over lossy network.
- Detectors (sensors) are placed in each lane at the upstream and downstream direction for counting and event triggering.
- Left and right lanes have sensors to count the vehicles going in these directions.
- Estimated time to travel from one intersection to another with 80 Km/h is around 45 sec.
- Each intersection operates in 4 phase's mode with parallel movements as default, which means that every two parallel directions will run at the same time.

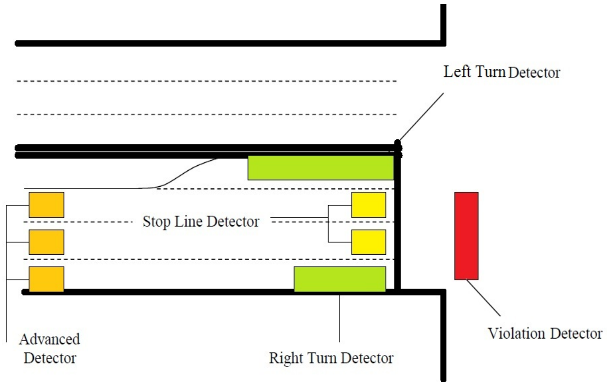


Figure 4.29: Vehicle Detectors

- The average of arrivals for each parallel direction will be taken as input, and the considered phases are (N, S,E,W,NW, SE,EN,WS).
- Simulation runs for 30 minutes.
- In the simulation, we considered 5 intersections as shown in (Fig. 4.14) and we tried to compare the results from the proposed approaches.

From the estimator side, the simplest approach to model vehicle arrivals is to assume a uniform arrival. This will results in a deterministic, uniform arrival pattern which means constant time headway between all vehicles. However, this assumption is usually unrealistic, as vehicle arrivals typically follow a random process. Thus, a model that represents a random arrival process is needed and the most suitable one is the Poisson distribution with arrival rate of λ . In general, the car arrival is part of the queuing model (e.g. $M/M/1$ or $M/G/1$) which simulates the traffic signal operations. Basically the queue model is any service station with the following:

- One or multiple servers
- waiting area or buffer

The time τ_n is inter arrival time between cars n and $n+1$ and it is a random variable. The traffic light system is following the stochastic process behavior.

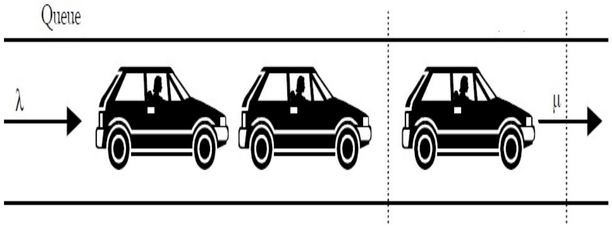


Figure 4.30: Basic Queue System

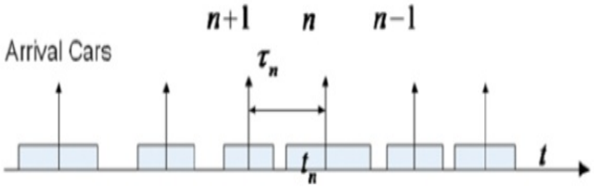


Figure 4.31: Cars Arrivals in time

The level of information exchanged is shown in Table 4.1. When we look to Fig.4.33, we can see that at intersection 1, we started with phase 1, then by the time the flow will reach to intersection 2, which is around 45 seconds, the incoming flow plus the existing flow will move together without stoppage and same will happen at intersection 3, this explanation is shown clearly in Fig.4.34. That shows the beauty of Quasi Decentralized approach over the Decentralized itself as shown in Fig.4.35, where in the Quasi we have benefited from the limited communication over a network to smooth and maximize the flow in certain direction between intersections.

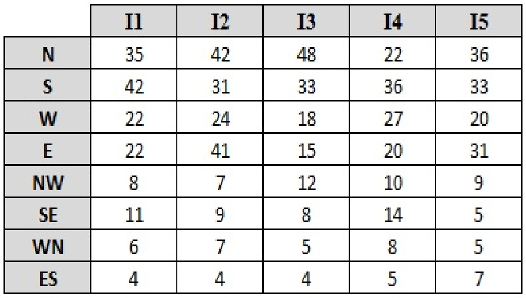
In DecNCS approach, the controller will control each intersection independently from others and the only information sent over lossy link is the arrival traffic via the sensors placed at the beginning of the roads towards that intersection. In the case of QuasiNCS, the information about the phase selection in each intersection is exchanged among the adjacent controllers, the one before and the one after, to allow continuous progression of platoons through successive signals along multiple intersections with minimum number of stops and sometimes

-	Dec	Quasi	Dist	Hir
Traffic Queues	Y	Y	Y	Y
$Phase\ Selection(I_i to I_j)$	N	Y	Y	Y
$Traffic\ Arrival(I_i to I_j)$	N	N	Y	Y
$GreenTime(in I_j)$	N	N	Y	Y
Traffic Jam Info	N	N	N	Y
Avg arrivals speed	N	N	Y	Y

Table 4.1: Data exchange in each approach

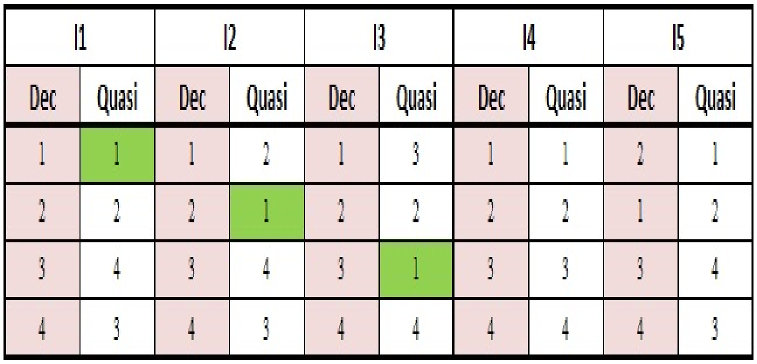
without stopping based on the traffic density, because the on/off nature of traffic signals tends to accumulate the vehicles in longer queue. The total trip time in the case of DecNCS will be more than 135 sec to cross the distance starting from intersection 1 to 3 with 2 stoppages while in QuasiNCS it is around 94 sec with no stoppage. Also, we can observe from Fig.4.35 that a synchronization can happen between intersection 2 , 4 and 5 where the majority of the traffic between East and West can run smoothly in the successive intersections.

In the DecNCS, the data transmission between sensors to controllers are over lossy network, so in case there is packets delay or dropout or another communication constraints, the controller can depend only on the last received data and in the case of long failure of the sensors due to physical damage, the controller can depend either on fixed green time (45 sec) or will be based on the average arrival rate computed from historical data. Another option is to sue a pre-timed table. For the QuasiNCS, if the phase selection information will be affected by any delay or dropout, it will simply run based on the arrival of the actual data coming from the sensors.



	I1	I2	I3	I4	I5
N	35	42	48	22	36
S	42	31	33	36	33
W	22	24	18	27	20
E	22	41	15	20	31
NW	8	7	12	10	9
SE	11	9	8	14	5
WN	6	7	5	8	5
ES	4	4	4	5	7

Figure 4.32: A Sample of Car Arrivals Rate /Min (Q)



I1		I2		I3		I4		I5	
Dec	Quasi	Dec	Quasi	Dec	Quasi	Dec	Quasi	Dec	Quasi
1	1	1	2	1	3	1	1	2	1
2	2	2	1	2	2	2	2	1	2
3	4	3	4	3	1	3	3	3	4
4	3	4	3	4	4	4	4	4	3

Figure 4.33: Decentralized and Quasi Phase Selection

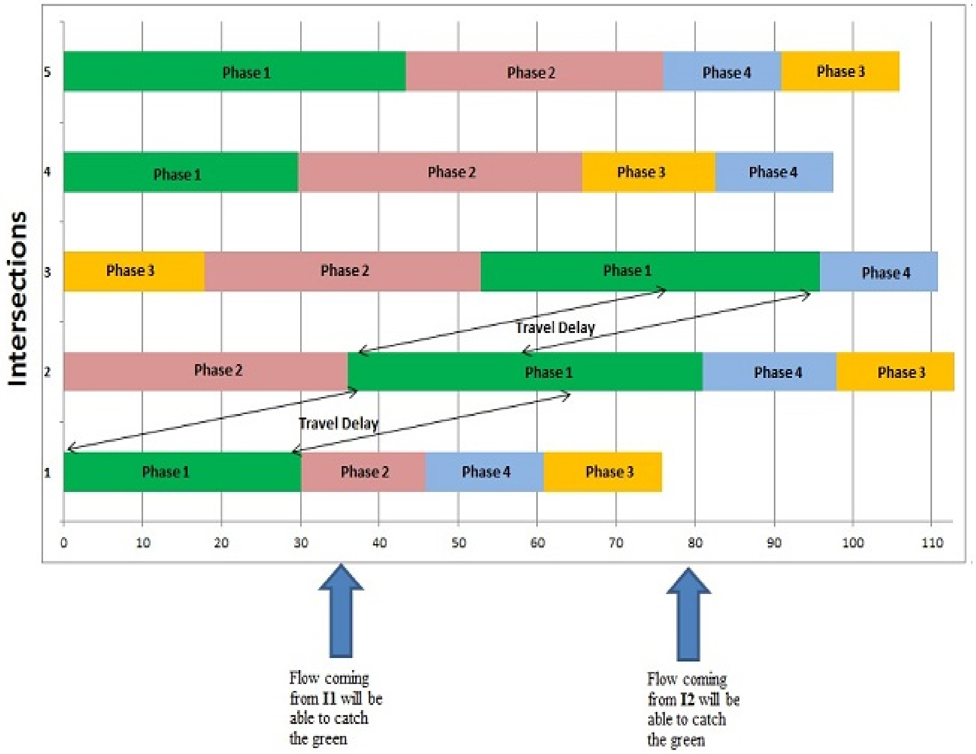


Figure 4.34: Phase Selection in QuasiNCS for Each Intersection to maximize the flow from intersection 1 up to 3

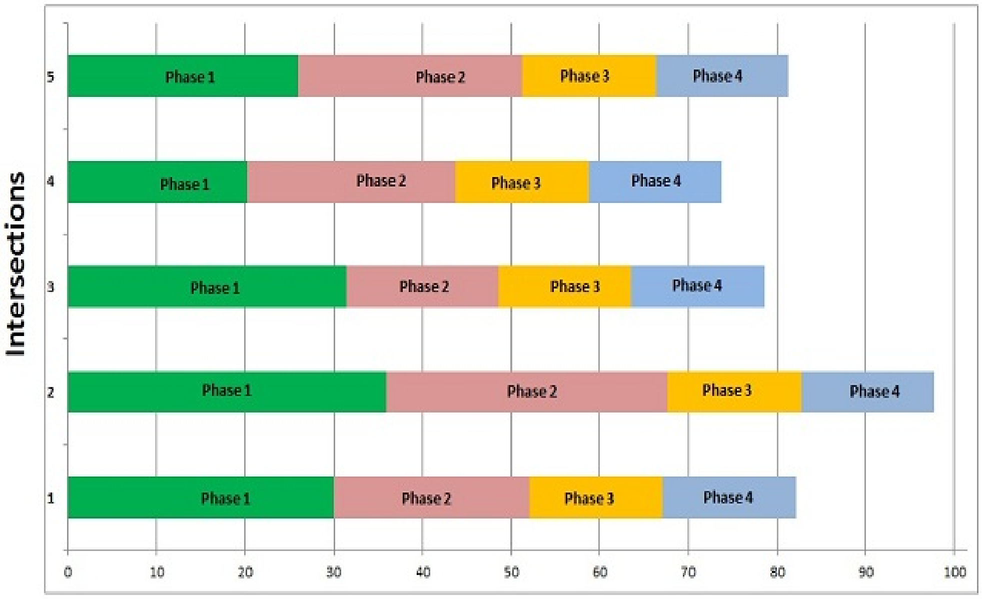


Figure 4.35: Phase Selection in DecNCS for Each Intersection

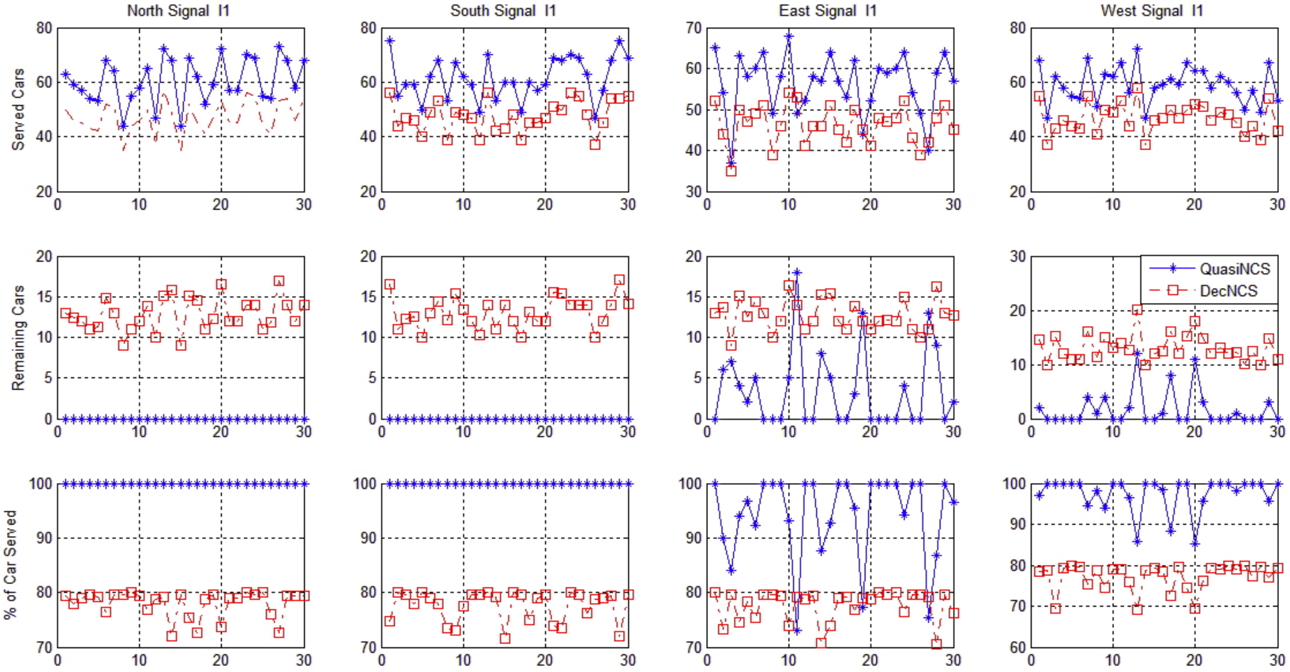


Figure 4.36: QuasiNCS vs. DecNCS , Intersection 1 , Main Directions

More simulations results are also presented in the coming figures and several scenarios for simulation results when traffic the density < 1 and $>= 1$, as shown in Fig. 4.36 up to Fig. 4.46. During all these simulation we have random exponential value for β, random delays, poisson distributed, and communication constraints are also enabled.

Fig. 4.46 shows several information about the traffic during the simulation of QuasiNCS.

The other figures, namely from Fig. 4.47 - 4.51, show several information about the traffic during the simulation of QuasiNCS (number of served cars vs. car arrival, and the remaining cars not served in that cycle), with normal traffic Density (< 0.5), and it is clearly showing that with the QuasiNCS, the ”served cars” signals track very well the ”car arrivals” signals, which indicate a very smooth traffic as expected.

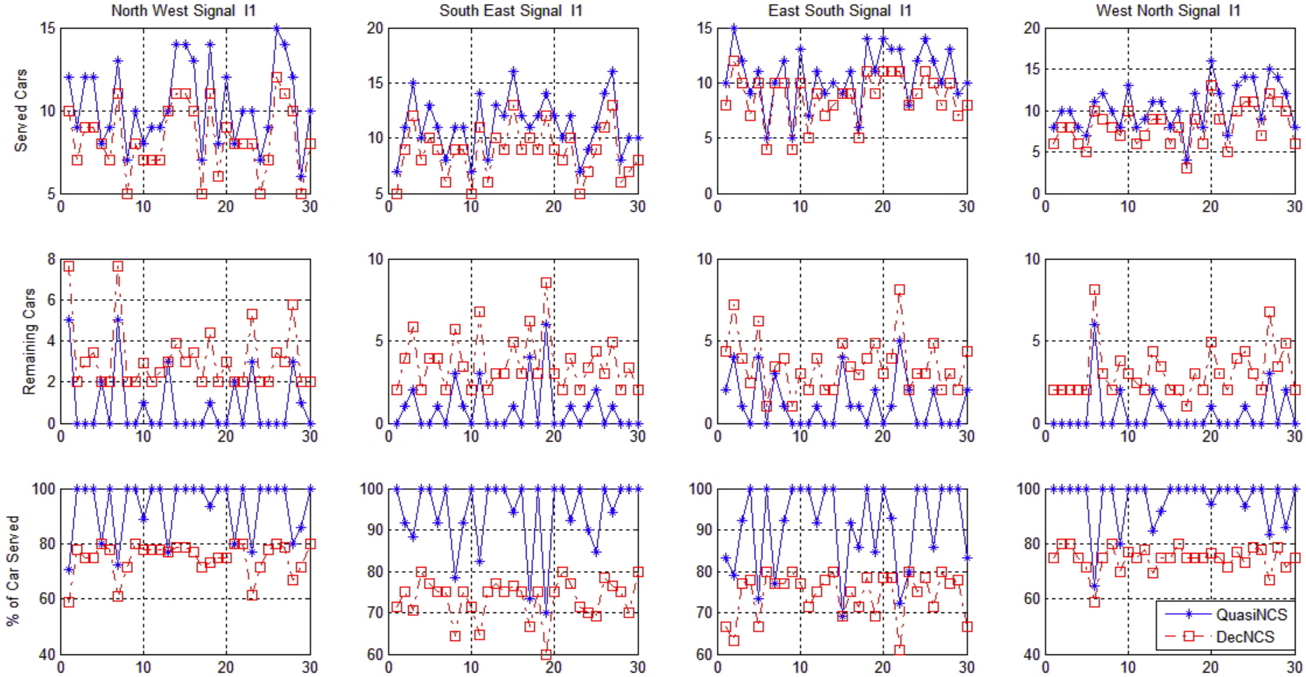


Figure 4.37: QuasiNCS vs. DecNCS , Intersection 1 , Sub Directions

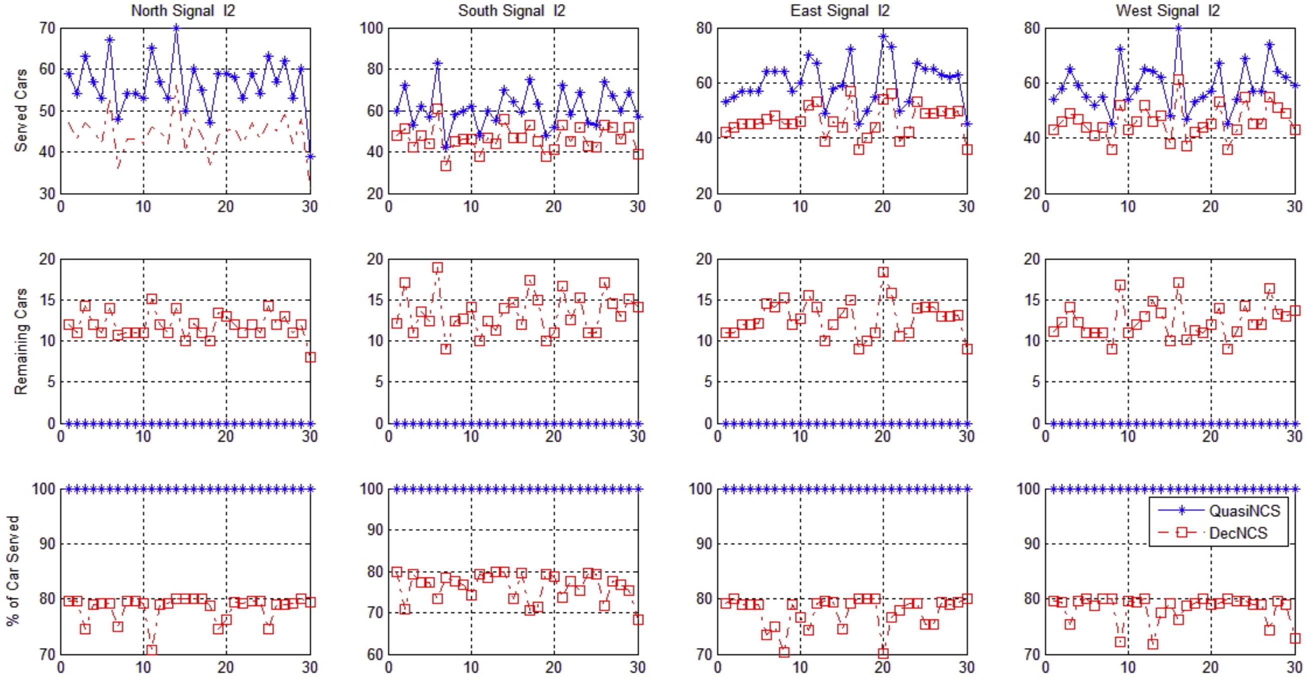


Figure 4.38: QuasiNCS vs. DecNCS , Intersection 2 , Main Directions

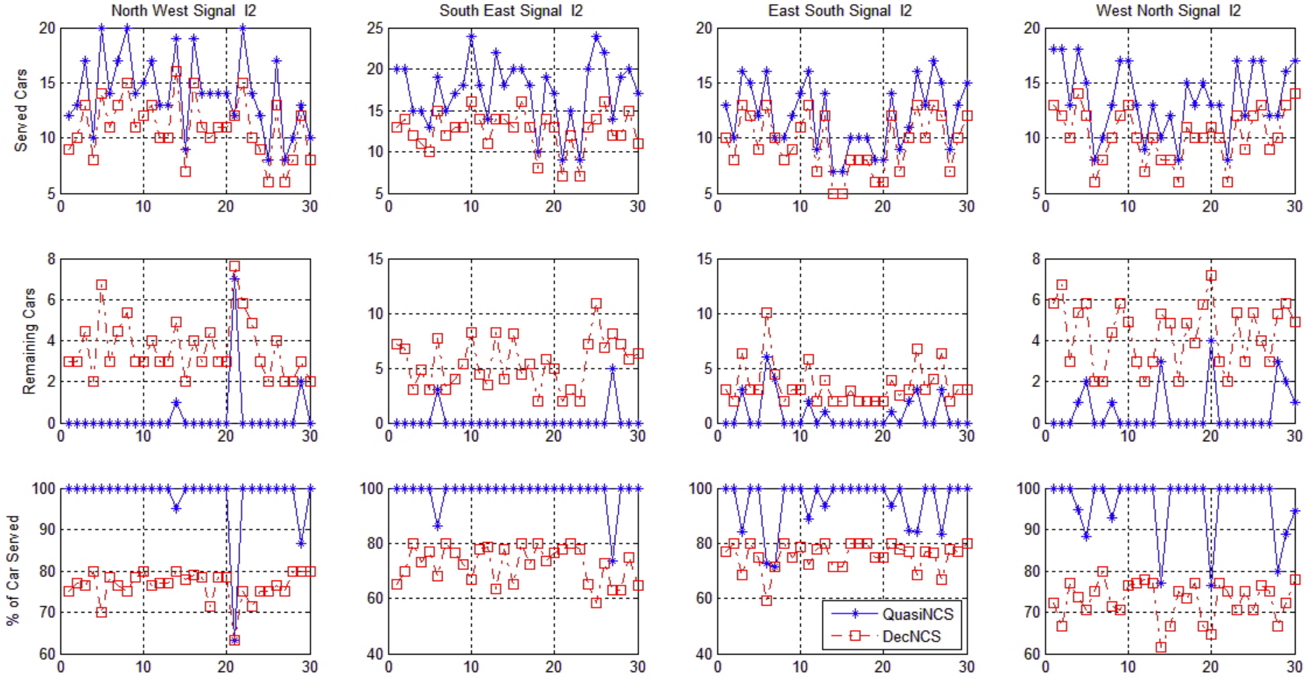


Figure 4.39: QuasiNCS vs. DecNCS , Intersection 2 , Sub Directions

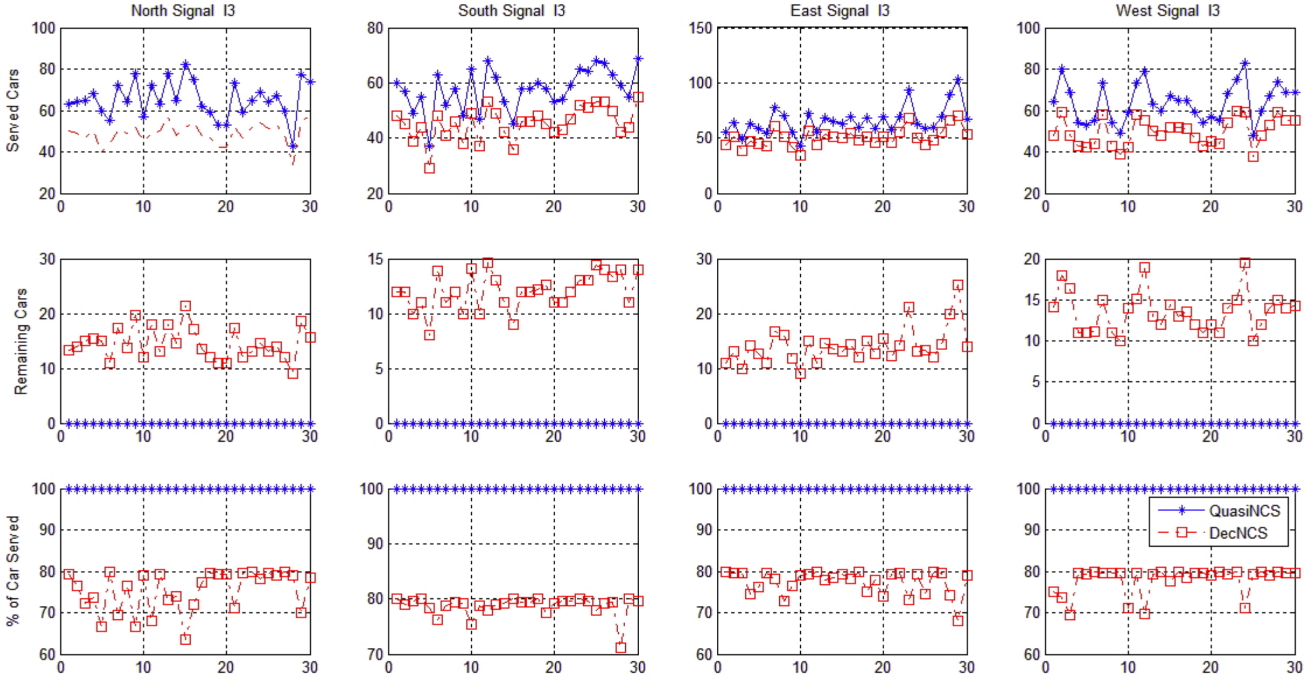


Figure 4.40: QuasiNCS vs. DecNCS , Intersection 3 , Main Directions

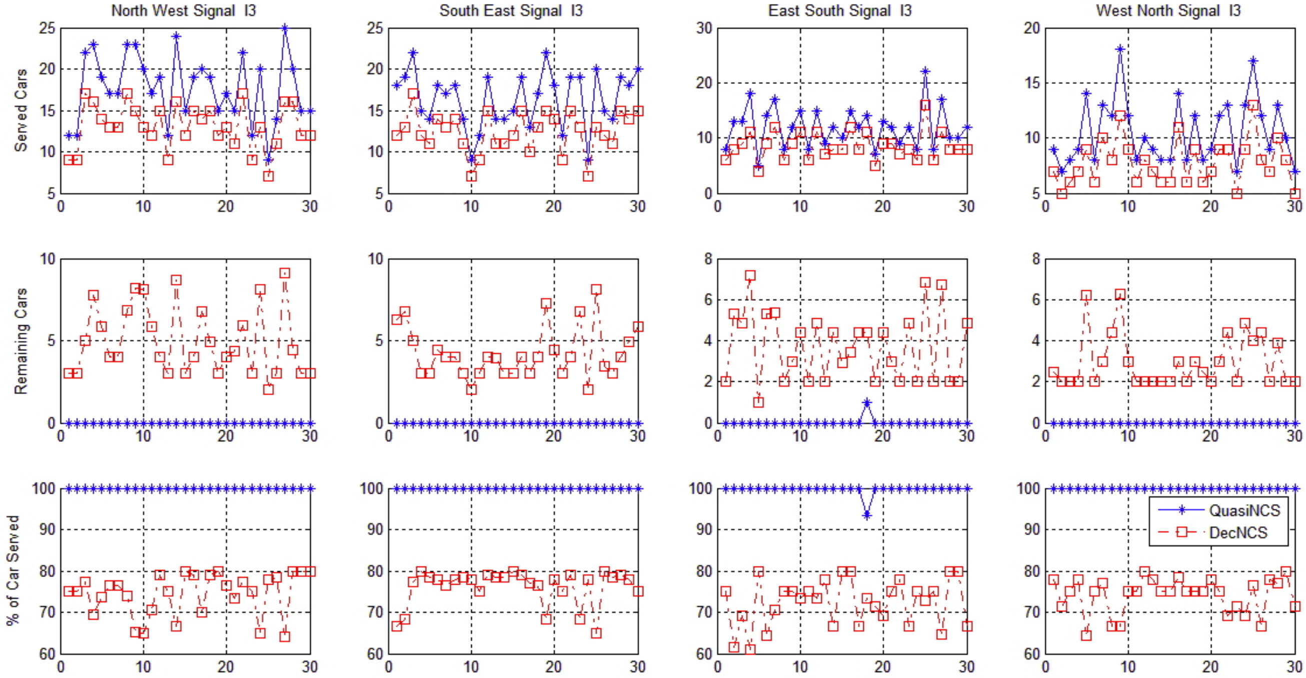


Figure 4.41: QuasiNCS vs. DecNCS , Intersection 3 , Sub Directions

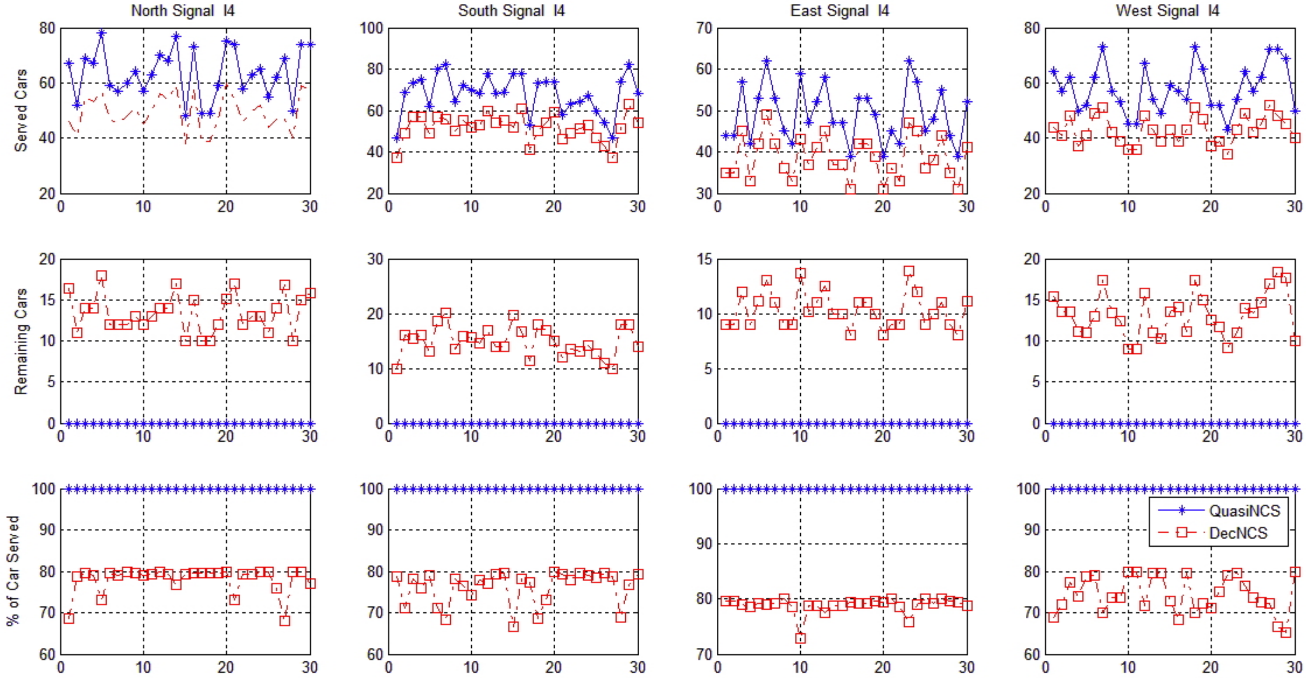


Figure 4.42: QuasiNCS vs. DecNCS , Intersection 4 , Main Directions

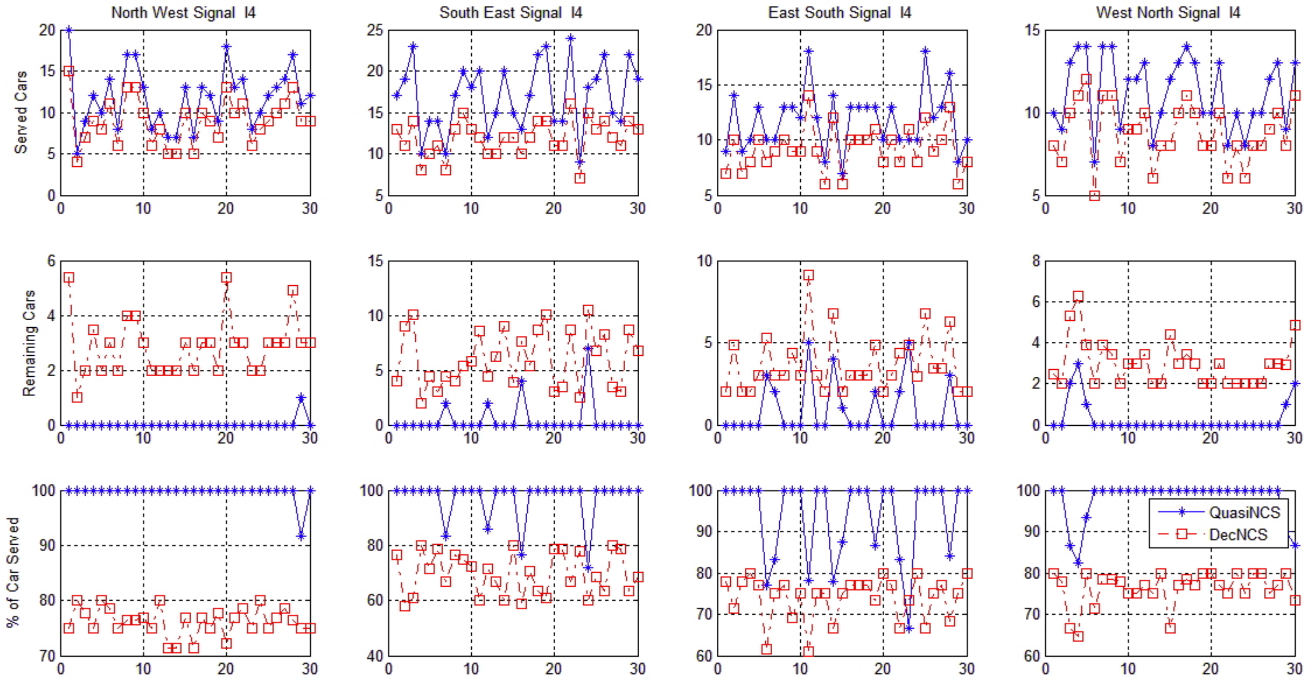


Figure 4.43: QuasiNCS vs. DecNCS , Intersection 4 , Sub Directions

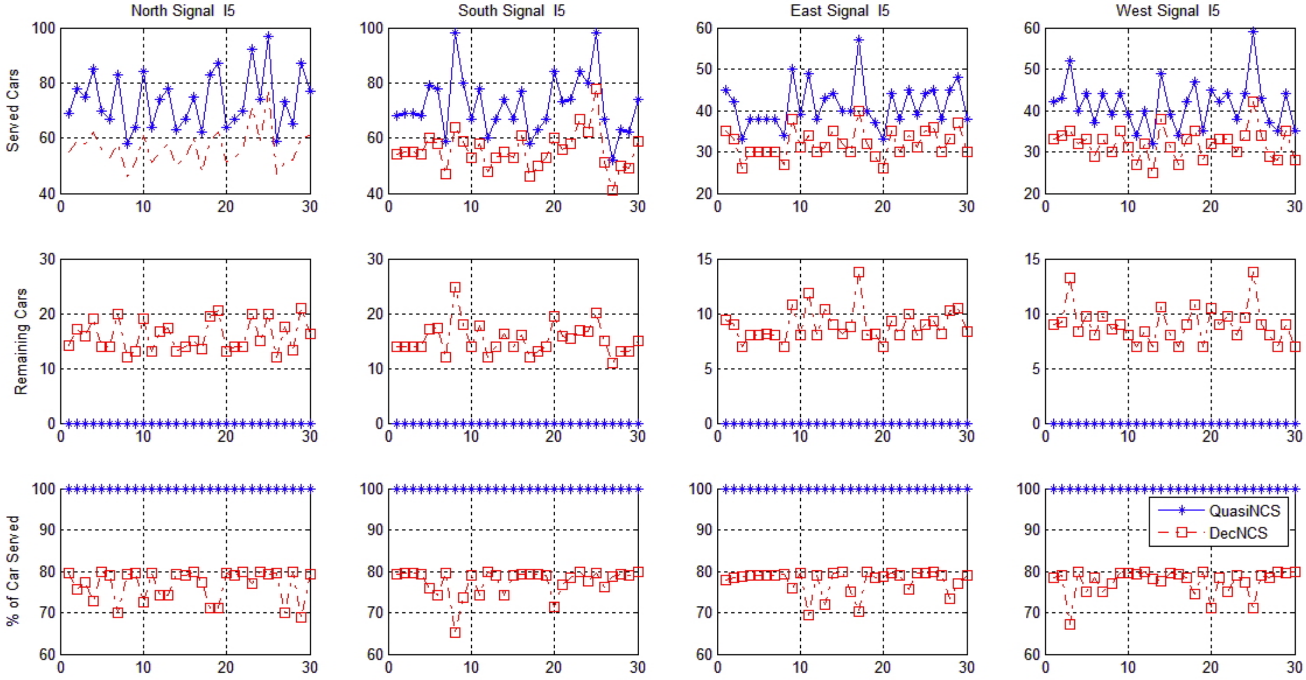


Figure 4.44: QuasiNCS vs. DecNCS , Intersection 5 , Main Directions

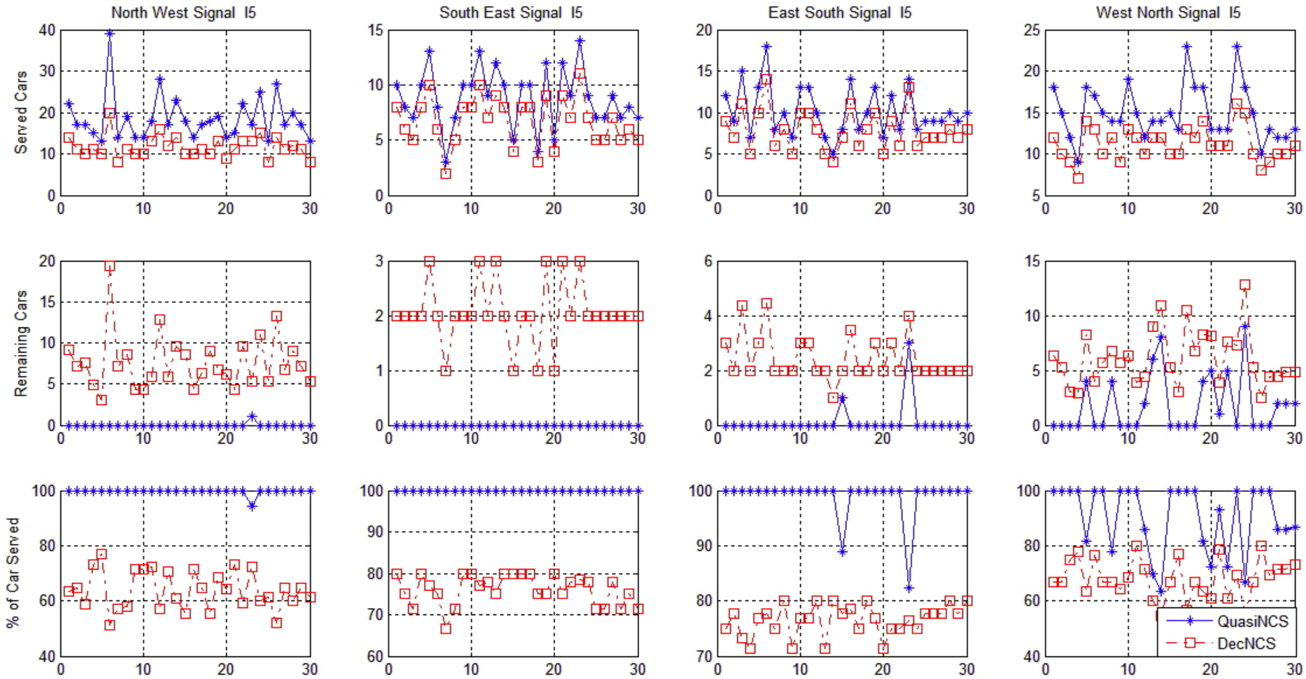


Figure 4.45: QuasiNCS vs. DecNCS , Intersection 5 , Sub Directions

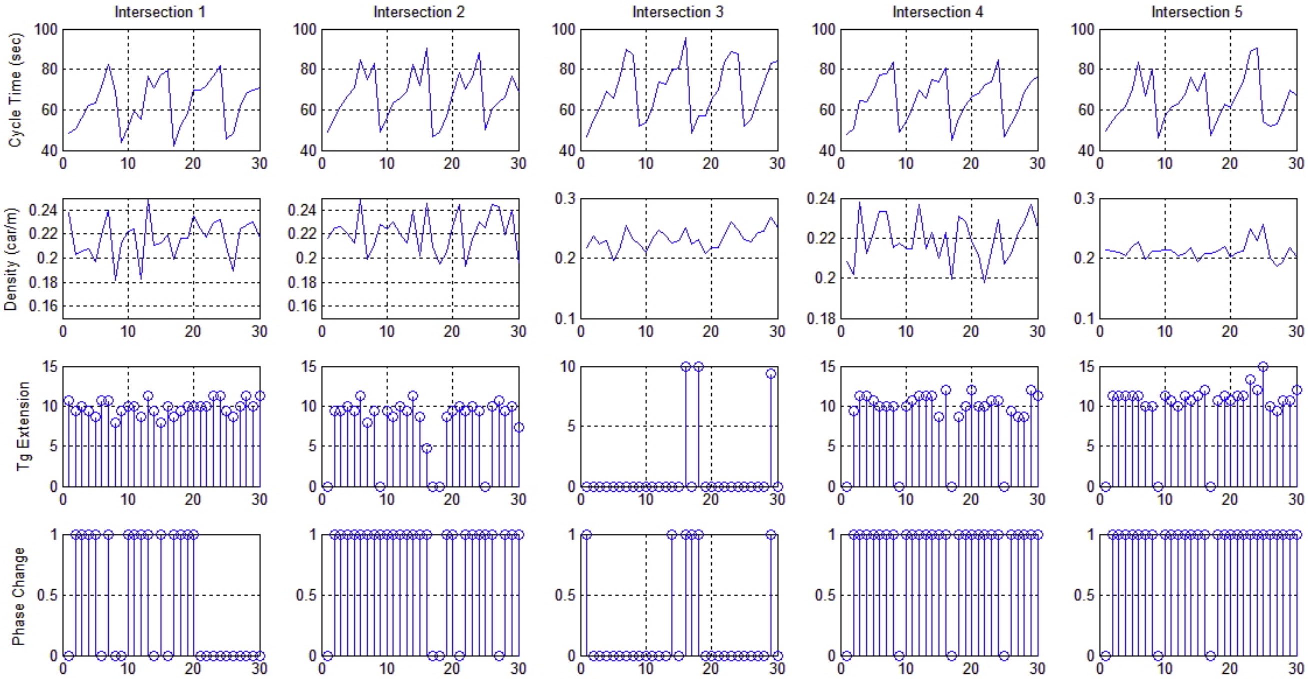


Figure 4.46: QuasiNCS Traffic Data

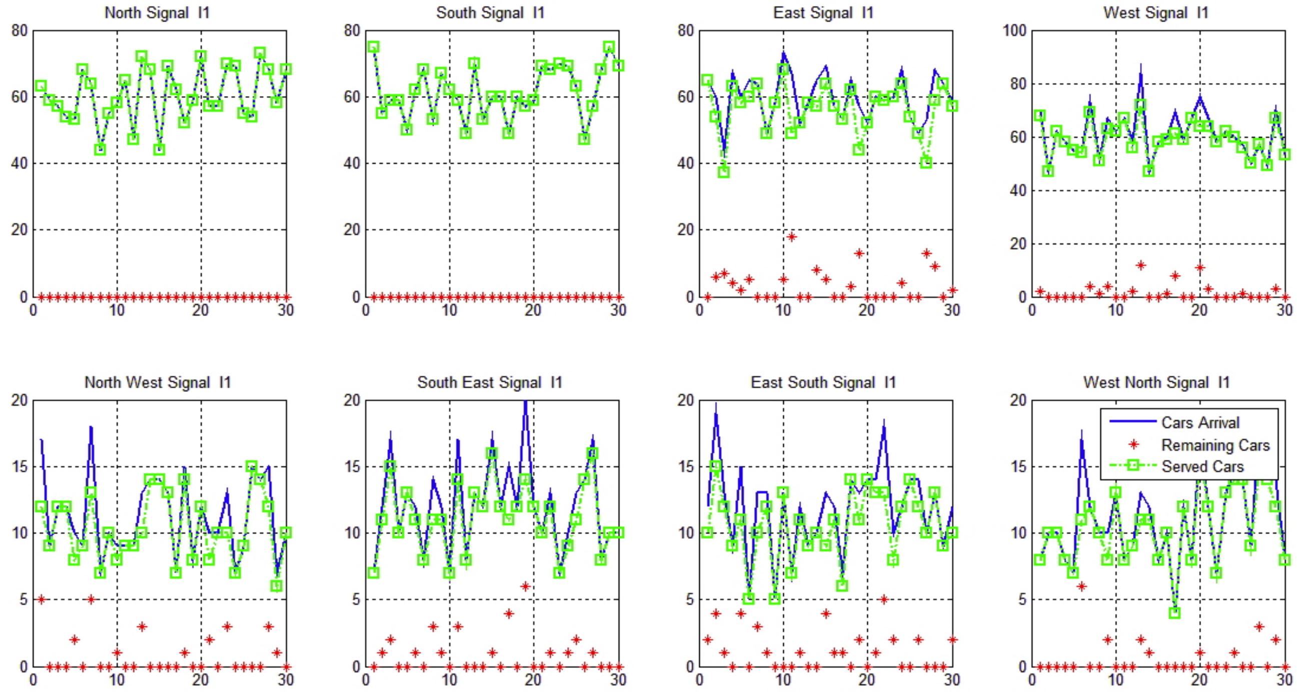


Figure 4.47: QuasiNCS Served vs. Arrived Cars , Intersection 1

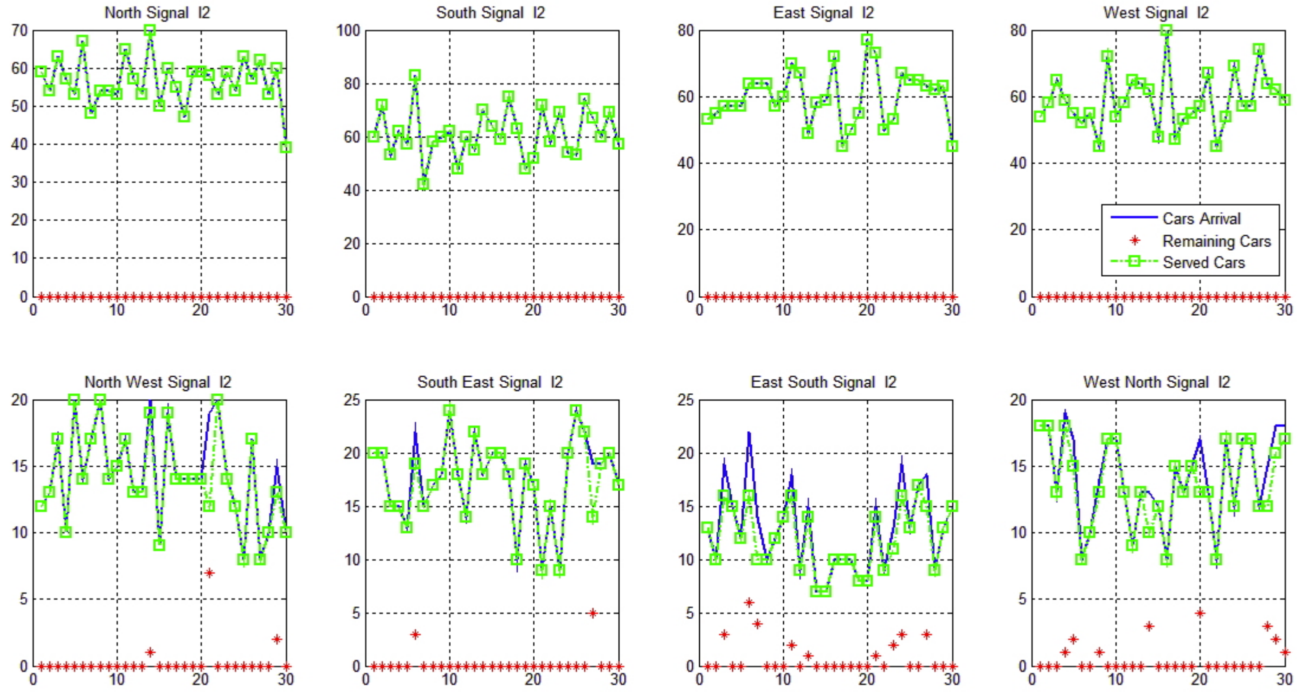


Figure 4.48: QuasiNCS Served vs. Arrived Cars , Intersection 2

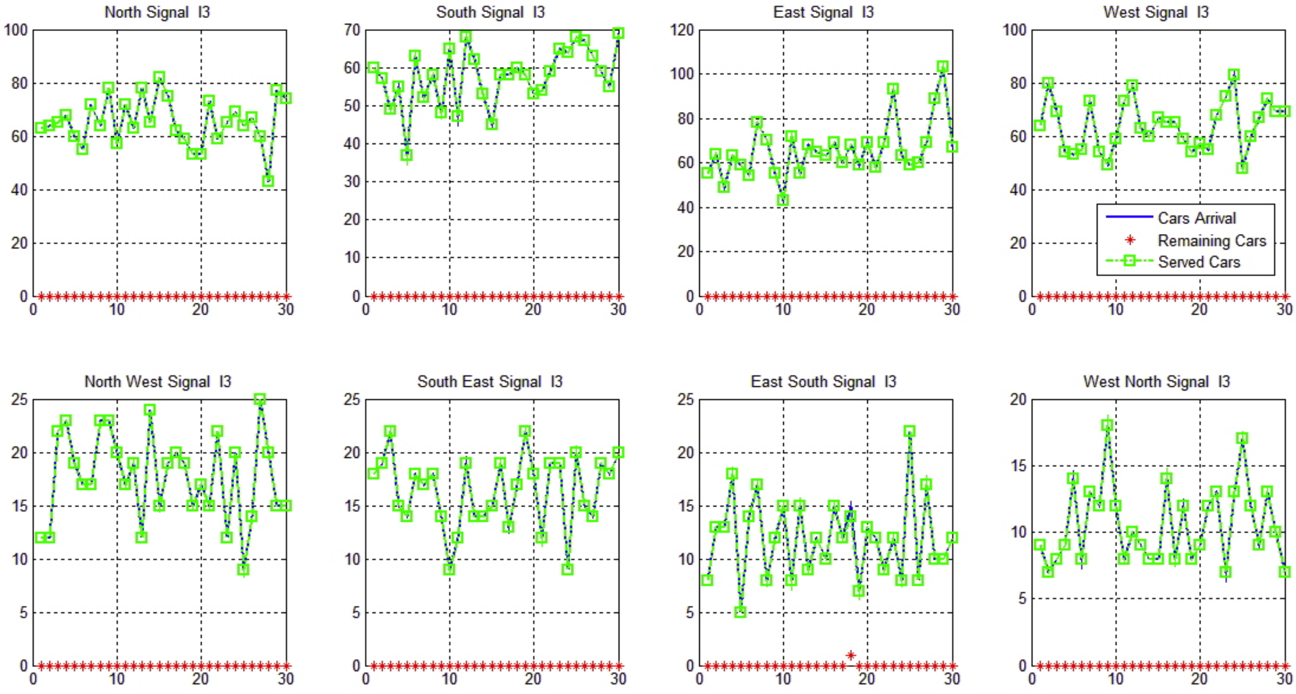


Figure 4.49: QuasiNCS Served vs. Arrived Cars , Intersection 3

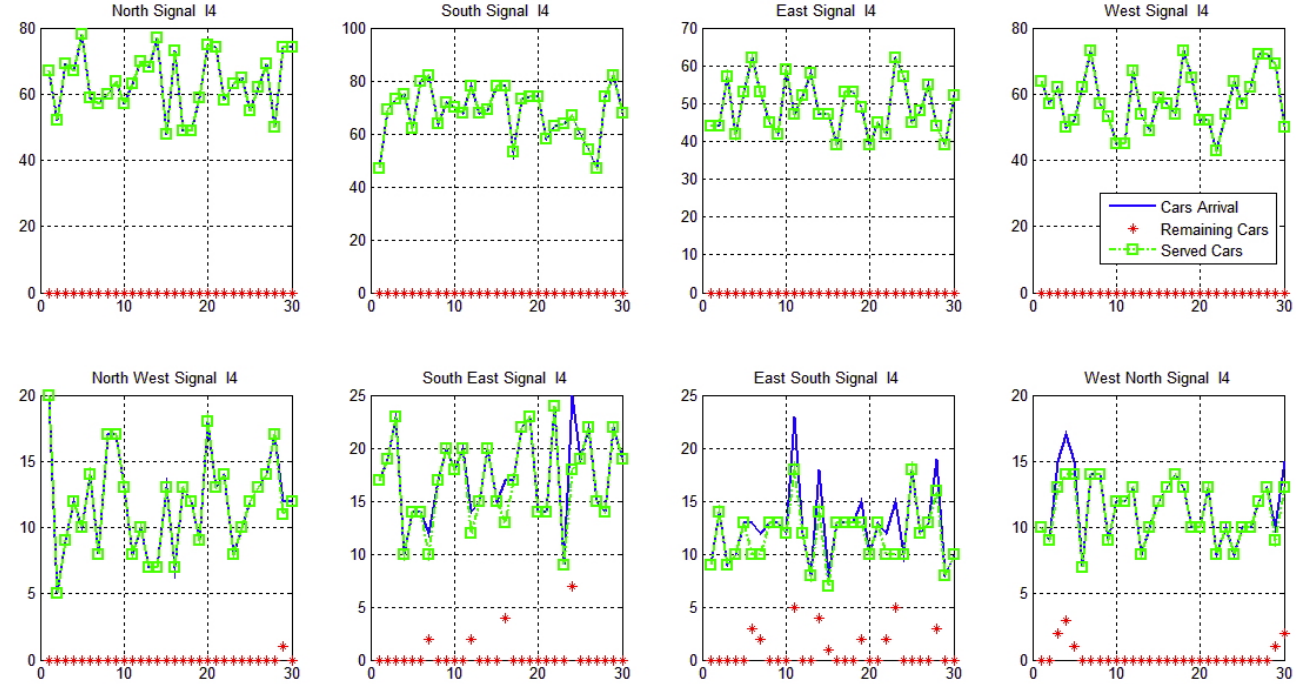


Figure 4.50: QuasiNCS Served vs. Arrived Cars , Intersection 4

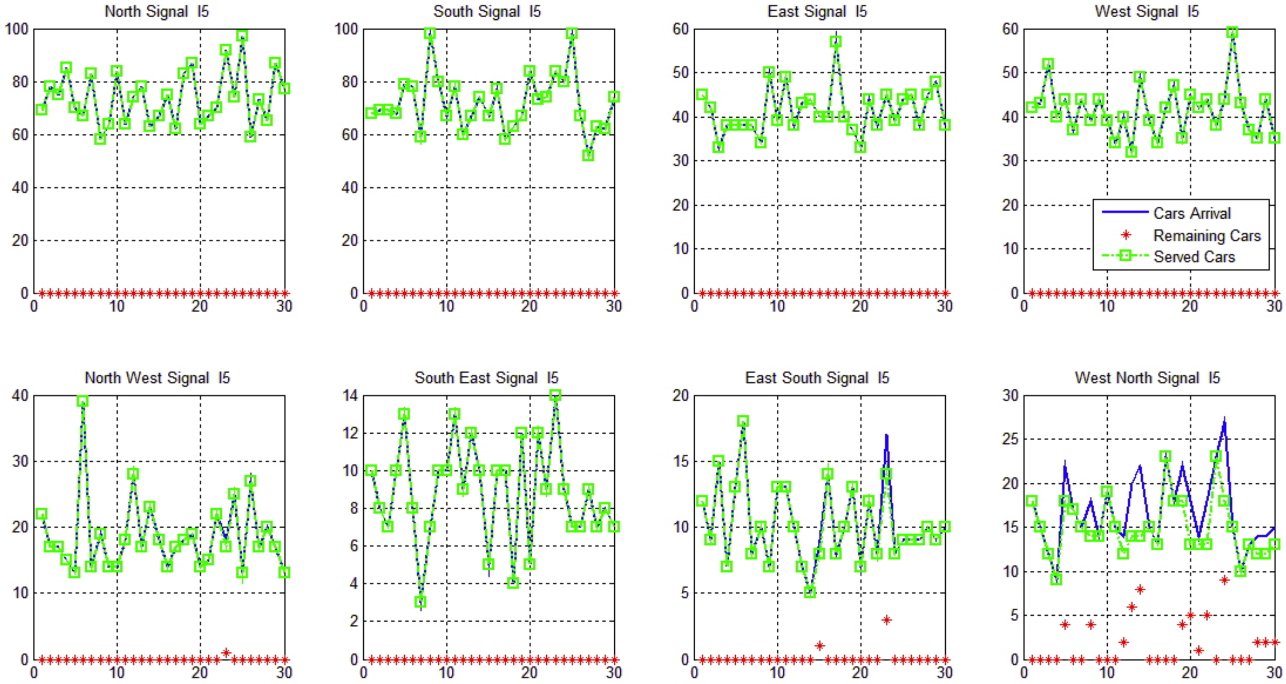


Figure 4.51: QuasiNCS Served vs. Arrived Cars , Intersection 5

4.14.1 More Simulation Results

In this section we will highlight some important simulation results in the following list:

- Communication Constraints: communication is an important factor for the controller to make the proper coordinations with other controllers. In the case of DecNCS, there is no communication between controllers and decision will be made on the intersection data only. From the first look, you may see that the DecNCS is giving low cycle time but in reality it is much more because it did not consider the new arrivals. For the QuasiNCS, we have simulated the effect of communication constraints as shown in Fig. 4.52, you can see that the more communication we allowed, the more the cycle time changes and this is required for the proper coordination between intersections considering the current and new coming traffic for each intersection.

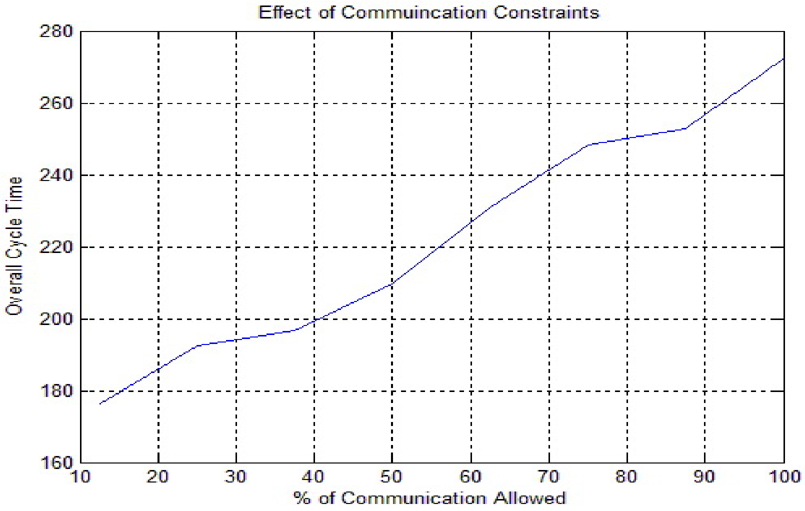


Figure 4.52: Communication Constraints Effect

- Packets Dropout: another issue we can show also is the effect of β values (sensor packets dropout, see eq. 3.16) which will affect more the QuasiNCS as shown in Fig. 4.53, and the dropout of sensors packets will reflect on the cycle time but not too much because usually such sensors applications will send few packets (number of cars, time, ...etc) cyclically, and if the packet dropout is increasing, the controller will switch to the local intersection control because may be the sensors are malfunction or physically damaged.

- Computation Time: the traffic density is not really an issue for the computation time as we can see from Fig. 4.54.

- Waiting Time: this is very important measure for the control system, because the longer the waiting time the more the drivers will get frustrated and the potential of violation will be higher. So, in this simulation, we focused on the waiting time behaviour during an incremental traffic by increasing the traffic arrival every cycle by 25% and we stop increasing it when the

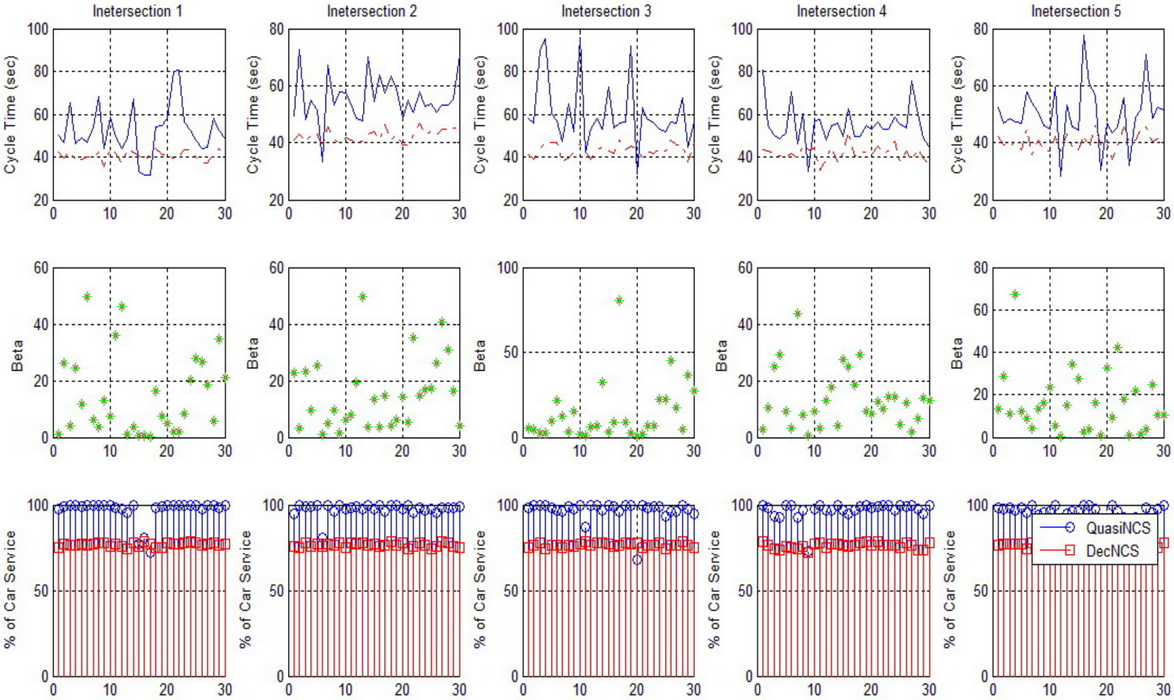


Figure 4.53: Beta Values Effect

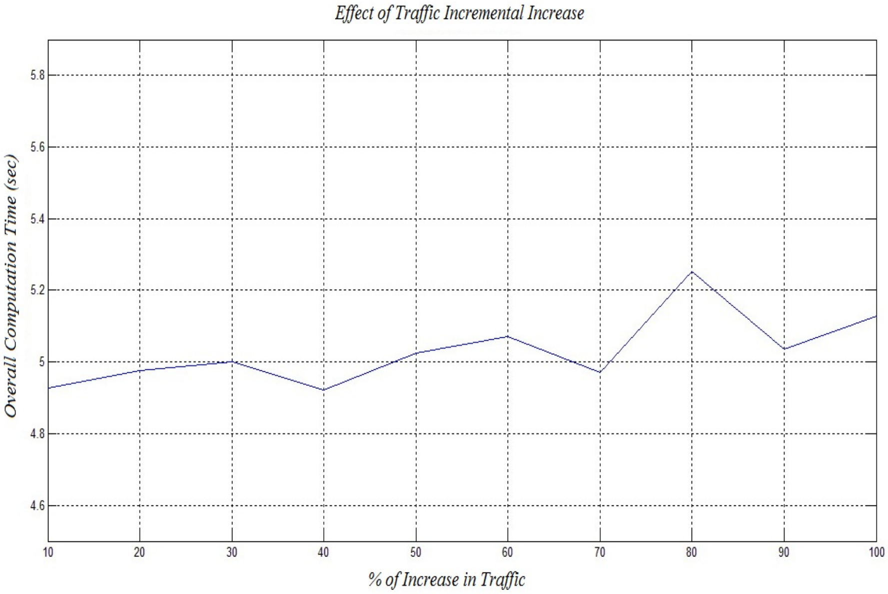


Figure 4.54: Traffic Density Effect on Control Computation Time

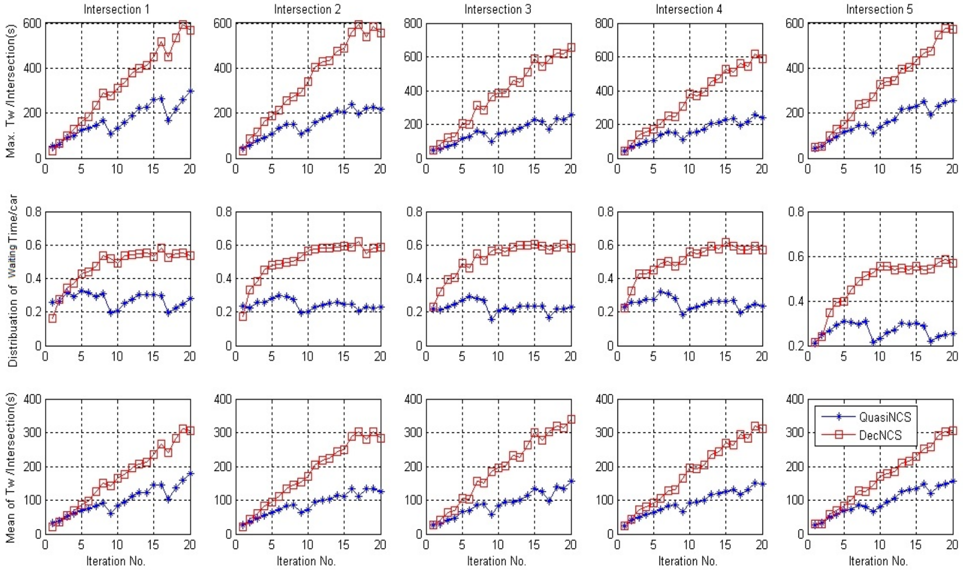


Figure 4.55: Incremental Traffic Density Effect on Waiting Time

traffic density exceeds 1 as shown in Fig. 4.55 and Fig. 4.56 shows the waiting time during normal random arrival.

- Communication Delay: the effect of data packets delay from previous intersection controller to the next intersection controller will let the $2nd$ controller to increase the intersection cycle time to accommodate the incoming traffic up to a certain limit then it will not extend. If the packets delay exceeded the maximum allowed limit, then the controller will ignore the delayed packets and start a new control cycle and if this problem continues for certain number of cycles, which means that the link needs a longer time to be fixed, the controller will then use one of the options we mentioned earlier (Historical data, Fixed Time or behave like DecNCS locally), Fig. 4.57 explains this issue clearly. Also, we can see from same figure in Intersection 2 after certain time it will stop doing green time extension because the delay exceeded the limit, similarly we can observe with intersection 4 and 5.

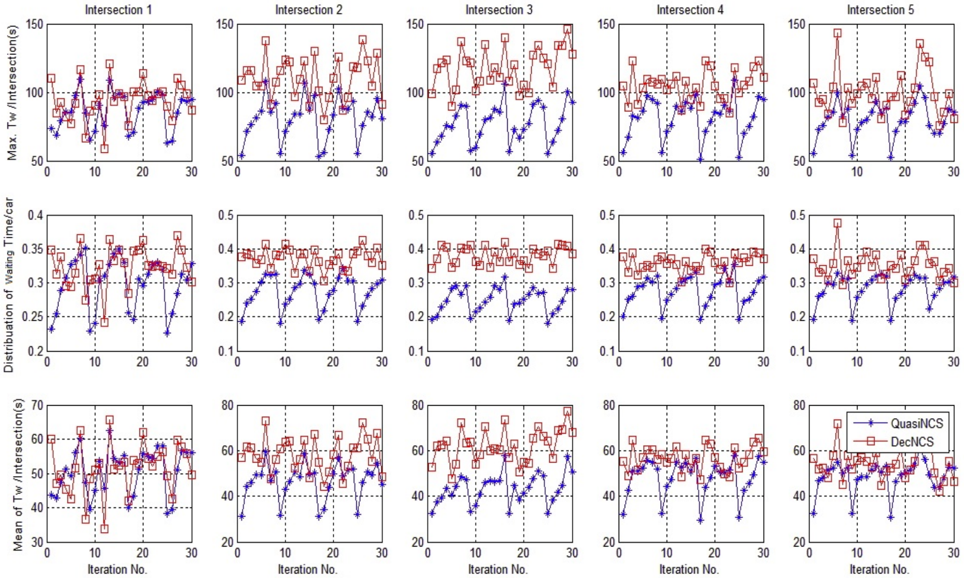


Figure 4.56: Random Traffic Density Effect on Waiting Time

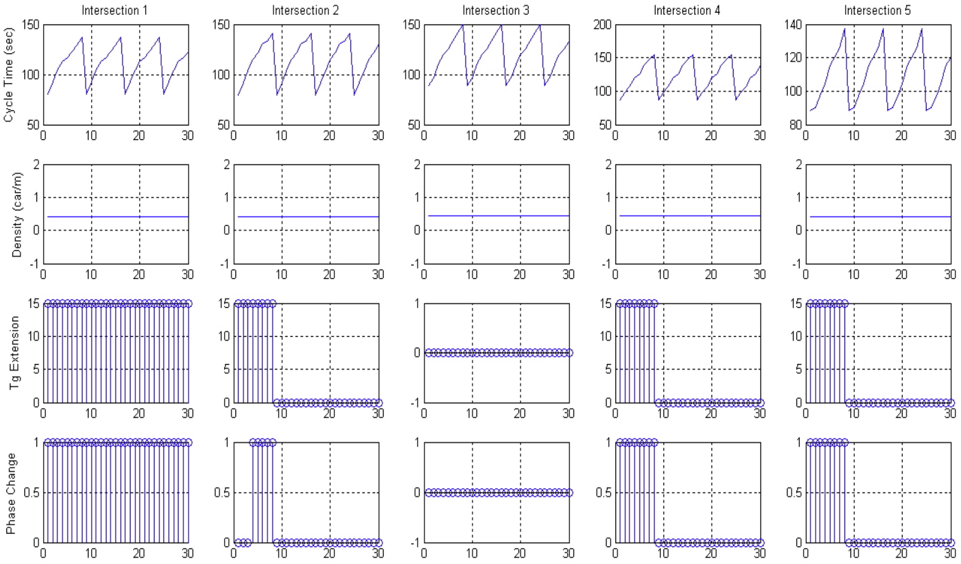


Figure 4.57: Delay Effects on Cycle Time

4.15 Complexity Study

Computational complexity is the study of the complexity of problems that is, the difficulty of solving them. Problems can be classified by complexity class according to the time it takes for an algorithm usually a computer program to solve them as a function of the problem size. Some problems are difficult to solve, while others are easy. For example, some difficult problems need algorithms that take an exponential amount of time in terms of the size of the problem to solve. Computational complexity may be approached from many different aspects. It can be investigated on the basis of time, memory or other resources used to solve the problem. Time and space are two of the most important and popular considerations when problems of complexity are analyzed.

Even though a problem may be computationally solvable in principle, in actual practice it may not be that simple. These problems might require large amounts of time or an inordinate amount of space. Also, there exist a certain class of problems that although they are solvable in principle they require so much time or space that it is not practical to attempt to solve them. These problems are called intractable. There is another form of complexity called hierarchical complexity. It is orthogonal to the forms of complexity discussed so far, which are called horizontal complexity

Since we are talking about three different control approaches, hence, each one has different system complexity due to different factors. We can list down these factors as follows:

1. LMI Size: The equations in (4.11-4.25) clearly express the LMI size of each approach and it is smaller for the DecNCS and the largest is DNCS. This is expected because in DecNCS we eliminate all types of communication between systems and basically each system is working alone without the

-	DecNCS	QuasiNCS	DNCS
LMI Size	(nxn)	3(nxn)	6(nxn)

Table 4.2: LMI Size for each approach

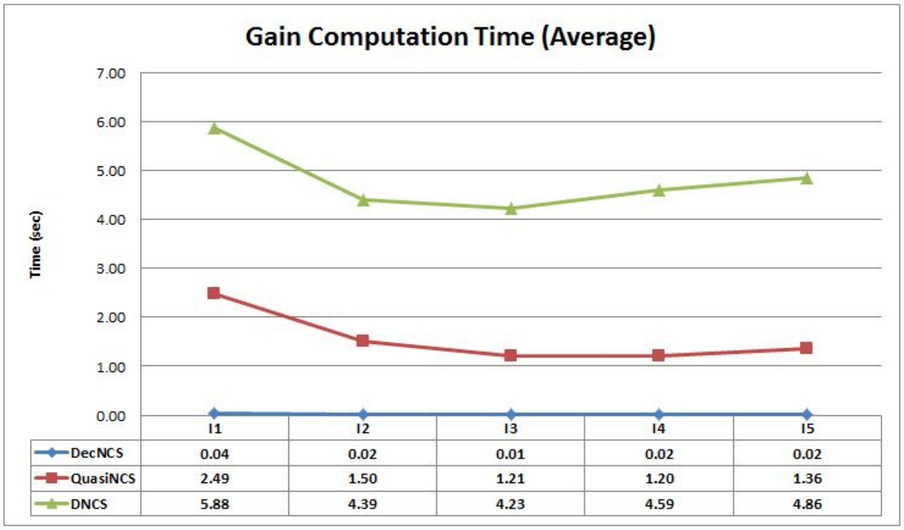
knowledge about others. For the QuasiNCS we allowed minimum communication which help at least the neighbours to coordinate between them at least. The case in DNCS is the most complex where all communications are allowed which adds overhead on the communication channel. Table 4.2 briefly shows the LMI size for each approach.

2. Gains Computations Time: the CPU time required for calculating the required gains and making the decision for each intersection is very small in the case of DecNCS compared to DNCS while in QuasiNCS it is reasonable and it is not as long as in the DNCS case. Figure 4.58 explain the CPU time under a heavy communication load for each direction in each intersection and, as a reminder, we have simulated 5 intersections , with 8 directions each. It is clearly that the DNCS is the most expensive approach while the QuasiNCS provides a good solution as we have seen in earlier sections with lower computation time (almost $\frac{1}{5}$) of DNCS. In normal load, we could get a lower computation time as shown in Fig. 4.59 but still the computation time for the DNCS remains too high compared to others. The values shown in Fig. 4.59 are the averages for each intersection over many simulation runs.

3. Lanes Characteristics: this factor basically depends on the physical road structure and changing it is not easy. Usually more lanes allow more cars to move at the same time when the signal is green. However, in our work the number of lanes is fixed for all approaches. An important remark is to keep the lane density less than the lane capacity or in other words, the ratio shall be < 1. Another lane characteristic is the lane width which is also fixed in

	I1	I2	I3	I4	I5
DecNCS	0.03	0.03	0.03	0.03	0.04
QuasiNCS	2.85	2.88	2.91	2.73	2.99
DNCS	9.28	9.96	9.84	10.02	9.94

Figure 4.58: Computation Time Under High Communication Load



	I1	I2	I3	I4	I5
DecNCS	0.04	0.02	0.01	0.02	0.02
QuasiNCS	2.49	1.50	1.21	1.20	1.36
DNCS	5.88	4.39	4.23	4.59	4.86

Figure 4.59: Average Computation Time under Normal Situation

all approaches in our work. The standard lane width is between 2.5 to 3.25 or 3.75 meters, based on the country standards. A single lane capacity of vehicles is simply the inverse of the tip-to-tip headway (S_h) and the most often expressed in vehicles/hour:

$$L_c = 3600/S_h \tag{4.31}$$

4. Communication Cost: this is clearly shown in the size of the LMI when we consider the QuasiNCS and DNCS that we have highlighted in point no.1 .
5. Speed Limit: if the speed limit is an input to the observer model, basically in DNCS, then it will require some more calculations to provide the suitable control. Generally, they use it for simulation purpose and for the realtime controller there will be sensors that send the speed measurements to the intersection controller for the required controller calculations. However, this type of information can be fed to the controller in case no speed sensors or speed data not arrived to help in reevaluating the green time selection.

CHAPTER 5

CONCLUSION AND FUTURE DIRECTIONS

5.1 Conclusions

In this thesis, we have carefully examined the decentralized, networked control architectures for interconnected dynamical systems. The work discussed the underlying rationale for the individual architectures and illustrated the fields of application and the merits/demerits as reported in the literature. Moreover, we have shown a single discretized system without any extra parameters. Then we added to it the intercommunication between systems in the decentralized design over network. After that we considered the networked-induced delays. Packet dropout and a varying sampling interval were included. Also, we have set a criterion to select the min sample interval and we defined a switch function for the transmission constraints.

Also, we have studied different control techniques like quasi-decentralized over network and we highlighted the major points about the distributed and hierarchal architectures over communication network. The models obtained were applied on a real life applications which is the Signalized Traffic Multi-Intersections Control

(STMIC) application. A comprehensive survey was done about the traffic control methods and techniques including several traffic concepts and fundamentals. The work was presented in state-space model and has considered several network parameters we have list earlier as a result of introducing the shared communication link into our control system.

In the simulation part, we performed the 1^{st} simulation on a standard or typical data to illustrate the theory obtained for DecNCS as first step in chapter 3 and shows that it works fine, where later on we can go ahead and extended a little that structure to achieve better results for more complicated applications . Then, simulation was more detailed for STMIC application where more experiments were done on multi signalized intersections control and coordination with different environments, control strategies and objectives (e.g. showing the effects of network, performance comparison between proposed control strategies, ...etc). Finally, discussion about complexity issues were included at the end of the chapter and it was clearly shown that the QuasiNCS, that requires minimum cross information, performed much better than DecNCS for traffic coordination and control application and in the same time it lower in complexity, computation time and resources than the DNCS.

5.2 Future Directions

The extension of this work has many directions specially the signal intersection traffic management problem that was addressed in the thesis.

- Optimization algorithms: Many optimization algorithms have been developed to solve efficiently the minimization problems related to linear and nonlinear centralized control. On the contrary, optimization methods for distributed and decentralized networked control are still lacking. This is an important and critical point where significant improvements are expected.

- Generalizing the traffic management model to wider range like city grids.
- Discussing specific communication media (e.g. wireless) and study the possible control techniques and solutions.
- Design a dynamical Graphical User Interface traffic simulator that allows interactive parameters modification during the simulation run to see the impacts on the traffic system.
- Reconfigurable control structures and hybrid systems: With reference to the hierarchical structures we should explore the possibility to reconfigure the system, for example by adding or removing actuators and sensors "plug and play control", see [57]-[59]. This could be useful to consider time varying performance requirements and to control systems described by a hybrid model. Finally, a flexible control configuration can better cope with the requirement of a high tolerance to faults.
- System partitioning: In the design of decentralized and distributed control the process under control, must be partitioned, if possible, a-priori into subsystems properly defined to reduce the dynamic couplings and to facilitate the control design. In some cases partitioning is natural in view of the process layout, see for example [63] for power grids and chemical plants in [64] are considered.
- Selection of the control structure: Criteria must be developed for the selection of the proper control structure based on the relative improvements achievable by increasing the complexity [8].
- Cover the uncertainties issues in the traffic control using the Robust control technique and Robust stability methods.

- Recently, some progress has also been made in solving the finite capacity stabilization problem for nonlinear systems [93], and for linear systems with unknown parameters [94]. Performance limitations of feedback over finite capacity memory-less channels are addressed in [95], which obtains a general extension of Bodes integral inequality [88]-[92].

.1 Linear Matrix Inequalities

Linear Matrix Inequalities (LMIs) methodology is a standard way to describe convex constraints in optimization problems. Optimization subject to LMIs is called semi-definite programming. LMIs are widely used in control because they appear naturally in many problems. Furthermore, there exist computationally efficient polynomial time algorithms such as interior point methods that can be applied easily to it. Therefore, semi-definite programming problems are always solvable in the sense that it can be determined whether or not the problem is feasible, and if it is, a feasible point that minimizes the cost function globally can be computed with a prespecified accuracy.

.1.1 A Brief History of LMIs in Control Theory

The history of LMIs in the analysis of dynamical systems goes back more than 100 years. The story begins in about 1890, when Lyapunov published his seminal work introducing what we now call Lyapunov theory. He showed that the differential equation:

$$\frac{d}{dt}(x(t)) = Ax(t) \tag{1}$$

is stable (i.e., all trajectories converge to zero) if and only if there exists a positive-definite matrix P such that

$$A^T P + PA < 0 \tag{2}$$

The requirement $P > 0$, $A^T P + PA < 0$ is what we now call a Lyapunov inequality on P, which is a special form of an LMI. Lyapunov also showed that this first LMI could be explicitly solved. Indeed, we can pick any $Q = Q^T > 0$ and then solve the linear equation $A^T P + PA = -Q$ for the matrix P, which is guaranteed to be positive definite if the system (1.1) is stable. In summary, the first LMI used to analyze stability of a dynamical system was the Lyapunov inequality (1.2), which can be solved analytically (by solving a set of linear equations).

to make it short, a summary of key events in the history of LMIs in control theory is the following:

- 1890: First LMI appears; analytic solution of the Lyapunov LMI via Lyapunov equation.
- 1940's: Application of Lyapunov's methods to real control engineering problems. Small LMIs solved "by hand".
- Early 1960's: PR lemma gives graphical techniques for solving another family of LMIs.
- Late 1960's: Observation that the same family of LMIs can be solved by solving an ARE.
- Early 1980's: Recognition that many LMIs can be solved by computer via convex programming.
- Late 1980's: Development of interior-point algorithms for LMIs.

It could be fair to say that Yakubovich is the father of the field, and Lyapunov the grandfather of it.

.1.2 LMI Matrices and Variables

A linear matrix inequality is an expression of the form:

$$F(x) = F_0 + \sum_{i=1}^{M} (F_i x_i) < 0 \tag{3}$$

where $[x_1 \, ..., \; x_n] \in R^n$ are decision variables and $F_i \in R^n$ is a set of symmetric matrices. In general, the LMI problems will not appear with the above form with scaler variables. Instead, we will encounter from now on LMIs with matrix variables. For example, consider the Lyapunov matrix inequality:

$$A^T P + PA < 0, P > 0 \tag{4}$$

where $P = P^T \in R^{n \times n}$ is the matrix variable. Generally, an LMI constraint with a matrix variables can be written as:

$$F(P_1 \, ..., \; P_m) \;\; = \;\; F_0 + \sum_{i=1}^{m} (U_i P_i V_i) < 0 \tag{5}$$

where $P_1 \, ..., \; P_m$ are the matrix variables, and U_i, P_i, V_i are given matrices.

.1.3 Standard LMI problems

*The LMI Problem:*It is the problem of determining whether a certain LMI is feasible or not,and if it is, to find one feasible point. It can be written as:

$$\begin{aligned} &\text{Find } x^* \\ \text{such that } &F(x^*) > 0 \end{aligned} \tag{6}$$

The Eigenvalue Problem It is the problem of minimizing the maximum eigenvalue of a matrix depending on a variable, or declaring that the problem is not feasible. It can be written as

$$\begin{aligned} &\text{minimize } \lambda \\ \text{subject to } (\lambda I - F(x)) > 0,\ \ &G(x) > 0 \end{aligned} \tag{7}$$

System of LMIs Several LMI constraints can be always casted into a single LMI. For example, $F_1(x) > 0, F_2(x) > 0$ can be written as:

$$\begin{pmatrix} F_1(x) & 0 \\ 0 & F_2(x) \end{pmatrix} > 0 \tag{8}$$

the following also are some LMIs important relations:

Congruence Transformation Consider $F > 0$, then $WFW^T > 0$ with W full rank. Therefore, we can always pre-multiply and post-multiply an LMI by a full rank matrix and its transpose.

Schurs Complement The Schurs complement is one of the most common ways for obtaining LMIs. It states that the pair of inequalities:

$$Q_1 - Q_2^T Q_3^{-1} Q_2 < 0 \tag{9}$$

$$Q_3 > 0 \tag{10}$$

which is equivalent to:

$$R = \begin{pmatrix} Q_1 & Q_2^T \\ Q_2 & Q_3 \end{pmatrix} > 0 \tag{11}$$

Change of Variables It is possible that by defining new variables to linearize some matrix inequalities. For example, consider synthesizing a state feedback control law $u_k = Kx_k$ to stabilize the system $x_{k+1} = Ax_k + Bu_k$. Using the Lyapunov inequality, we can write:

$$(A + BK)^T P(A + BK) - P < 0, P > 0 \tag{12}$$

which is a nonlinear inequality in P, K. Noting that $P = PP^{-1}P$, we can use Schurs complement to write the matrix inequality as:

$$\begin{pmatrix} P & (A+BK)^T P \\ P(A+BK) & P \end{pmatrix} > 0 \tag{13}$$

Define a new variable $Q = P^{-1}$, by multiplying both sides by the congruence transformation $diag[QQ]$, we get:

$$\begin{pmatrix} Q & Q(A+BK)^T \\ (A+BK)Q & Q \end{pmatrix} > 0 \tag{14}$$

Finally, we set $Y = KQ$ to get:

$$\begin{pmatrix} Q & QA^T + Y^T B^T \\ AQ + BY & Q \end{pmatrix} > 0 \tag{15}$$

which is an LMI in the variables Q, Y . We can get our original variables by $P = Q^{-1}$,$K = YQ^{--1}$.

.1 NEMA

NEMA is an acronym which stands for the **National Electrical Manufacturers Association**. This group develops standards and conventions for various pieces of traffic signal control equipment, including controllers and cabinets. The National Electrical Manufacturers Association is a trade association with 450 member organizations that sets standards for the generation, distribution, transmission, control and end-use of electricity. NEMA works in conjunction with the National Transportation Communications for Intelligent Transportation Systems Protocol to set standards governing traffic signals.

.1.1 A Brief History of NEMA

The older NEMA standard for traffic signal control equipment is known as the $TS - 1$ standard while the newer standard is known as the $TS - 2$ standard. The first version of the $TS - 1$ standard was introduced in 1975 and the first version of the $TS - 2$ standard was introduced in 1998. Prior to 1975, there was no industry standard for traffic control equipment and no interchangeability amongst controller manufacturers. As with traffic signal controllers, loop detector electronics units were developed and marketed by numerous manufacturers, each using a different type of harness connector and detection technique [209]. To overcome subsequent interchangeability problems, NEMA developed a set of standards known as "Section 7. Inductive-Loop Detectors". These were released early in 1981. This section of the NEMA Standards defined functional standards, physical standards, environmental requirements, and interface requirements for several inductive-loop electronics unit configurations. Section 7 described only the basic functions associated with inductive-loop detector electronics units. Users identified the need for additional functions for specific locations, particularly delay and extension timing. To cover this gap, NEMA developed and in 1983 released "Section 11. Inductive-Loop Detectors with Delay and Extension Timing." This section was basically identical to Section 7 with the addition of requirements for the timing of delayed call and extended call features. A further revision resulted in a new Section 15, which was released February 5, 1987. This new standard combines, updates, and supersedes Sections 7 and 11.

The NEMA Standards define two basic types of electronics unit configurations: shelf mounted and card-rack mounted. Shelf mounted units are commonly used in NEMA controllers and are available in both single-channel and multichannel (two- or four-channel) configurations. Outputs are generated by electromechanical relays or by electrically isolated solid-state circuits. Physical dimensions and connector requirements are included in the NEMA Standards. Card-rack mounted electronics units, fit into a multiple card rack and operate with external 24-volt DC power generated in the rack assembly or elsewhere in the controller cabinet. These devices are an effective way to reduce cabinet space requirements where large numbers of inductive-loop detector electronics units are needed. Still more standards can be found in NEMA about controllers, interfaces, detectors,...etc and the list shows few examples only:

- Presence and Pulse Modes of Operation.
- Timing Features.
- Tuning Range.
- Response Time.
- Operation with Grounded or Open Loops.
- Detector Terms and Definitions.
- Lightning Damage and Electrical Interference.

- Signal Light operations and specifications.
- Advanced Transportation Controller Specification (ATC).
- Environmental Requirements
- Load Switches
- Conflict Monitors
- Inductive Loop Detectors
- Flashers
- Signal Controllers
- .... etc

The last update of the NEMA $TS-1$ standard for traffic signal control equipment was published in 1989. It is still of interest since much of the traffic signal control equipment that exists along todays streets was installed under, and conforms to, this standard. The $TS-1$ publication provides standards on a variety of important topics, including: The NEMA $TS-2$ standard expands on the older NEMA TS1 Traffic Control Systems standard. The $TS-1$ standard was based on the philosophy that controllers would provide a basic set of features and standard connectors. Manufacturers would compete based on the hardware and software they provided inside the controllers. The NEMA $TS-1$ standard was successful for isolated actuated intersection control, but it lacked sufficient detail for implementing more advanced features, such as coordinated-actuated operation and preemption. Type 1 systems include the controller unit, conflict monitor, and the included features of each. Individual vendors supplemented the $TS-1$ standard by providing the complement of features necessary for deploying coordinated-actuated traffic signal systems. This introduced incompatibility and procurement issues, particularly when government agencies needed to upgrade existing signal systems at a later date and had to solicit competitive bids. Nevertheless, the competitive market forces continued to rapidly advance the state of the practice and created a following that led many States to adopt the NEMA standard. In the late 1980s and early 1990s, the NEMA $TS-1$ specification was updated with NEMA $TS-2$ to provide coordinated-actuated operation, preemption, and an optional serial bus to simplify wiring.

REFERENCES

[1] A. Al-Hammouri, M. Branicky, V. Liberatore and S. Phillips. "Decentralized and Dynamic Bandwidth Allocation in Networked Control Systems", *IEEE International Parallel and Distributed Processing Symposium*, pp. 8–18, 2006.

[2] L. Bakule. "Decentralized Control: An Overview", *Annual Rev. Control*, vol. 32, pp. 87–98, 2008.

[3] Chen Peng and Dong Yue. "State feedback controller design of networked Control systems with parameter uncertainty and State-delay". Asian Journal of Control, 8, pp. 385–392, 2006.

[4] L. Schenato. "Optimal Estimation in Networked Control Systems Subject to Random Delay and Packet Drop". *IEEE Transactions on Automatic Control*, vol.5, pp. 1311-1317, 2008.

[5] W. Zhang, M. S. Branicky and S. M. Phillips. "Stability of Networked Control Systems", *IEEE Control Systems Magazine*, pp. 88–99, 2001.

[6] K. Ji and W. Kim. Real-Time Control of Networked Control Systems via Ethernet, *Int. J. Control, Automation, and Systems*, vol.3, pp. 591–600, 2005.

[7] N. J. Ploplys, P. A. Kawka and A. G. Alleyne. "Closed Loop Control over Wireless Networks". *IEEE Control Systems Magazine*, vol.24, pp. 58–71, 2004.

[8] P. A. Kawka, A. G. Alleyn. "Stability and Feedback Control of Wireless Networked Systems". *American Control Conference*, 4, pp. 2953-2959, 2005.

[9] R. Scattolini. "Architectures for Distributed and Hierarchical Model Predictive Control A review". *J. of Process Control*, vol.19, pp. 2723–731, 2009.

[10] Y. Sun, N. H. El-Farra. "Quasi-decentralized Model-based Networked Control of Process Systems". Computers and Chemical Engineering, pp. 2016–2029, 2008.

[11] H. Cuiand and E. Jacobsen. "Performance Limitations in Decentralized Control". *Journal of Process Control*, vol.12, pp. 485–494, 2002.

[12] M. Baldea, P. Daoutidis, and A. Kumar. "Dynamics and Control of Integrated Networks with Purge Streams". *The American Institute of Chemical Engineers Journal*, 52, pp. 1460-1472, 2006.

[13] J.B. Rawlings, B.T. Stewart. "Coordinating Multiple Optimization-based Controllers: New Opportunities and Challenges". *Journal of Process Control*, vol.18, pp. 839-845, 2008.

[14] A. Rantzer. "Using Game Theory for Distributed Control Engineering". 3rd World Congress of the Game Theory Society, 2008.

[15] A. Gattami and B. Bernhardsson. "Minimax Team Decision Problems", American Control Conference, pp. 766-771, 2007.

[16] K. Rohloff, J. Ye, J. Loyall and R. Schantz. "A Hierarchical Control System for Dynamic Resource Management". *IEEE Real-Time and Embedded Technology and Applications Symposium*, 2006.

[17] B. T. Jevtovic and M. R. Matausek. "A Predictive-Adaptive Hierarchical Control System of Bucket-Wheel Excavator: Theory and Experimental Results" .SER.: ELEC. ENERG., vol.18, pp. 493–503, 2005.

[18] F.N. Bailey, M. Brdys, K. Malinowski and P. Tatjewski. Wozniak, Control and Coordination in Hierarchical Systems, Wiley and Sons, 1980.

[19] F. Amigoni and N. Gatti. "A formal framework for connective stability of highly decentralized cooperative negotiations". Autonomous Agents and Multi-agent Systems, vol.15, pp. 253–279, 2007.

[20] M. A. Brdys, M. Grochowski, T. Gminski, K. Konarczak, M. Drewa. "Hierarchical predictive control of integrated wastewater treatment systems". Control Engineering Practice, vol.16, pp. 751–767, 2008.

[21] E.J. Van Henten and J. Bontsema. "Time-scale decomposition of an optimal control problem in greenhouse climate management". Control Engineering Practice, vol.17, pp. 88–96, 2009.

[22] Robert W. Brennan and William O. "A Simulation Test-Bed to Evaluate Multi-Agent Control Of Manufacturing Systems". *Proc. 32nd Winter Simulation Conference*, pp. 1747–1756, 2000.

[23] J. Colandairaj, G. W. Irwin and W. G. Scanlon. "Analysis and Co-Simulation of an IEEE 802.11b Wireless Networked Control System". *Irish Signals and Systems Conference*, pp. 35–36, 2006.

[24] L. Schenato, M. Franceschetti, K. Poolla, and Shankar Sastry, "Foundations of Control and Estimation over Lossy Networks", *Proc. of the IEEE*, pp. 163–187, Jan. 2007.

[25] Q. Ling and M. Lemo, "Optimal Dropout Compensation in Networked Control Systems", *Proc. IEEE Conf. on Decision and Control*, vol. 1, pp. 670–675. December 2003.

[26] B. Marieke, G. Cloosterman, N. Wouw, W. H. Heemels, and H. Nijmeijer. "Stability of Networked Control Systems with Uncertain Time-Varying Delays". *IEEE Transactions on Automatic Control*, vol.54, pp. 1575–1580, 2009.

[27] S. Oh and S. Sastry. "Approximate Estimation of Distributed Networked Control Systems". *Proc. American Control Conference*, pp. 997–1002, 2007.

[28] Q. Zong, W. Liu and X. He. "Fault Diagnosis of Distributed Networked Control Systems with Stochastic Communication Logic". Intelligent Control and Automation, pp. 6877–6881, 2008.

[29] S. Oh and S. Sastry. "Distributed Networked Control System with Lossy Links: State Estimation and Stabilizing Communication Control". *45th IEEE International Conference on Decision and Control*, pp. 1942–1947, 2006.

[30] X. W. Lemmon, M.D. "Asymptotic Stability in Distributed Event-triggered Networked Control Systems with Delays". *Proc. American Control Conference*, pp. 1362–1367, 2010.

[31] X. W. Lemmon, M.D. "Event-Triggered Broadcasting across Distributed Networked Control Systems. *American Control Conference*, pp. 3139-3144, 2006.

[32] R. Gupta, A. Agarwal, M. Chow, W. Wang. "Characterization of Data Sensitive Wireless Distributed Networked Control Systems". *IEEE Int. conference on Advanced Intelligent Mechatronics*, ASME, pp. 1–6, 2007.

[33] M. Cloosterman, N. Wouw, M. Heemels, and H. Nijmeijer. "Robust Stability of Networked Control Systems with Time-varying Network-induced Delays". *Proc. 45th IEEE Conference on Decision and Control*, pp. 4980–4985, 2006.

[34] M. S. Mahmoud Decentralized Control and Filtering in Interconnected Dynamical Systems, CRC Press (Taylor and Francis Group), New York, 2010.

[35] M. S. Mahmoud. Decentralized Systems with Design Constraints, Springer-Verlag, UK, 2011.

[36] M. S. Mahmoud. Decentralized Reliable Control of Interconnected Systems with Time-Varying Delays, *J. Optimization Theory and Applications*, vol.143, pp. 497-518, 2009.

[37] F. Lian, J. Moyne, and D. Tilbury. "Network Design Consideration for Distributed Control Systems". *IEEE Transactions on Control Systems Technology*, vol.10, pp. 297–307, 2002.

[38] C. Robinson and P. Kumar. "Networked Control Systems with Packet Delays and Losses". *Proc. 47th IEEE Conference on Decision and Control*, pp. 4602–4607, 2008.

[39] J. Baras, X. Tan, and P. Hovareshti. "Decentralized Control of Autonomous Vehicles". *Proc. 42nd IEEE Conference on Decision and Control*, vol.2, pp. 1532–1537, 2003.

[40] M. C. Gennaro and A. Jadbabaie. Decentralized Control of Connectivity for Multi-Agent Systems. *IEEE Conference on Decision and Control*, pp. 3628–3633, 2006.

[41] N. Bauer, M. Donkers, W. Heemels and N. Wouw, "An Approach to Observer-Based Decentralized Control under Periodic Protocols". *Proc. 29th Benelux Meeting on Systems and Control*, pp. 2125–2131, 2010.

[42] J. Wei, "Stability Analysis of Decentralized Networked Control Systems", *World Congress on Intelligent Control and Automation*, vol.7, pp. 5477–5482, 2008.

[43] S. Yksel and T. Basar, "Communication Constraints for Decentralized Stabilizability with Time-Invariant Policies", em IEEE Trans. Automatic Control, vol.52, pp. 1060–1066, 2007.

[44] S. Yksel and T. Basar, "Decentralized Multi-Controller Stabilizability over Communication Channels", *Proc. European Control Conference*, 2007.

[45] S. Yksel, O. C. Imer, and T. Basar, "Constrained State Estimation and Control over Communication Networks", *Proc. Conf. Information Sciences and Systems*, 2004.

[46] S.Yksel and T. Basar, "Optimal Signaling Policies for Decentralized Multi-controller Stabilizability Over Communication Channels". *IEEE Trans. Automatic Control*, vol.52, pp. 1969–1974, 2007.

[47] A. Matveev and A. Savkin, "Decentralized Stabilization of Linear Systems via Limited Capacity Communication Networks", *Proc. IEEE Conf. on Decision and Control*, pp. 1155–1161, 2005.

[48] C. Zhang and G. E. Dullerud, "Decentralized Control with Communication Bandwidth Constraints", *Proc. American Control Conference*, pp. 1711–1716, 2007.

[49] A. Matveev and V. Savkin, "Decentralized Stabilization of Networked Systems under Data-Rate Constraints", *Proc. of the 17th World Congress*, pp. 8034–8038, 2008.

[50] S. Yuksel and T. Basar, "Communication Constraints for Decentralized Stabilizability with Time-Invariant Policies", *IEEE Trans. Automatic Control*, vol. 52, pp. 1060–1066, 2007.

[51] Y. Sun and N.H. El-Farra. "Quasi-decentralized Networked Control of Process Systems". *American Control Conference*, pp. 1507–1512, 2008

[52] Y. Sun and N.H. El-Farra. "Resource-aware Quasi-decentralized Control of Nonlinear Plants over Communication Networks". *American Control Conference*, pp. 154–159, 2009.

[53] Y. Sun and N.H. El-Farra. "Quasi-decentralized Control of Process Systems Using Wireless Sensor Networks with Scheduled Sensor Transmissions". *American Control Conference*, pp. 3390–3396, 2009.

[54] Y. Sun and N.H. El-Farra. "Quasi-decentralized Scheduled Output Feedback Control of Process Systems Using Wireless Sensor Networks". *IFAC Symposium on Advanced Control of Chemical Processes*, 2009.

[55] K. Rohloff , R. Schantzand, Y. Gabay. "High-Level Dynamic Resource Management for Distributed, Real-Time Embedded Systems". *Proc. the summer computer simulation conference*, pp. 749-756, 2007.

[56] P. A. Kawka, and A.G. Alleyne. "Robust Wireless Servo Control Using A Discrete Time Uncertain Markovian Jump Linear Model". *Proc. of the American Control Conference*, vol. 3, pp. 733-742, 2007.

[57] C. Peng and D. Yue. "State feedback controller design of networked Control systems with parameter uncertainty and State-delay". *Asian Journal of Control*, vol. 8, pp. 385–392, 2006.

[58] T. Knudsen, K. Trangbk, C.S. Kallese. "Plug and Play Process Control Applied to a District Heating System". Proc. of the European Control Conference, pp. 336-341, 2009.

[59] Control IT, AC800M/C Control Functions Users Guide. ABB Automation Technology Products 3BSE 021 351 R201, 2001.

[60] L. Pernebo and B. Hansson. Plug and Play in Control Loop Design. ABB Automation Technology Products, 3BSE028758, 2001.

[61] R.Vadigepalli, F. J. Doyle III. "Structural Analysis of Large-scale Systems for Distributed State Estimation and Control Applications". Control Engineering Practice, vol.11, pp. 895-915, 2003.

[62] R. Vadigepalli, F. J. Doyle III. "A Distributed State Estimation and Control Algorithm for Plantwide Processes". *IEEE transactions on control systems technology*, vol.11, pp. 119-128, 2003.

[63] K. Menighed, C. Aubrun, and J. J. Yam. "Distributed State Estimation and Model Predictive Control of Linear Interconnected System: Application to Fault Tolerant Control", *IEEE International Conference on Control and Automation*, pp. 936-941, 2009.

[64] A. Abur. "Distributed State Estimation for Mega Grids". 15th Power Systems Computation Conference. pp. 1-5, 2005.

[65] J.B. Rawlings, B.T. Stewart. "Coordinating multiple optimization-based controllers: new opportunities and challenges". *Journal of Process Control*, vol.18, pp. 839-845, 2008.

[66] H. Chan and U. Ozguner. "Closed-Loop Control of Systems over a Communication Network with Queues". *Proc. American Control Conference*, vol.1, pp. 811-815, 1995.

[67] Y. Halevi and A. Ray. "Integrated communication and control systems, Part I: Analysis". *J. Dyn. Syst., Meas. Control*, vol. 110, pp. 367-373, 1988.

[68] J. Nilsson. Real-time control systems with delays. Ph.D. dissertation, Dept. Automatic Control, Lund Inst. Technol., Sweden, 1998.

[69] A. Albert. "Comparison of Event-Triggered and Time-Triggered Concepts With Regard To Distributed Control Systems". *Proc. Embedded World*, pp. 235-252, 2004.

[70] J. Nilsson, B. Bernhardsson, and B. Wittenmark. "Some topics in real-time control". *American Control Conference*, vol.4, pp. 2386-2390, 1998.

[71] M. Torngren. "Fundamentals of implementing real-time control applications in distributed computer systems". Real-Time Systems, vol.14, pp. 219-250, 1998.

[72] J. Nilsson, B. Bernhardsson, and B. Wittenmark. "Stochastic analysis and control of real-time systems with random time delays", *Automatica*, vol.34, pp. 57-64, 1998.

[73] W. Kim, K. Ji and A. Ambike. "Networked Real-Time Control Strategy Dealing With Stochastic Time Delays and Packet Losses". *Journal of Dynamic Systems, Measurement and Control.* vol.128, pp. 681-686, 2006.

[74] C. C. Chen, S. Hirche and M. Buss. "Sampled-data Networked Control Systems with Random Time Delay". The International Federation of Automatic Control, 2008.

[75] A. Mihael, C. Boldisor and V. Comnac. "Influence of Variable Time Delay on the Performances of Networked Control Systems". Interdisciplinary in engineering Scientific Bulletin of the PetruMaior University of TirguMures 5, pp. 56-61, 2009.

[76] M. H. Behnam. "Flexible Scheduling for Real Time Control Systems based on Jitter Margin". Master Thesis, Mlardalens Hgskola, Vsters, 2005.

[77] B. Rahmani, A. Markazi and N. Mozayani. "Real Time Prediction of Time Delays in a Networked Control System". International Symposium on Communications, Control and Signal Processing, pp. 1242–1245, 2008.

[78] N. Vatanskib, J. Georges, Ca. Aubrun, E. Rondeaua and L. Jounelab. "Control Compensation Based on Upper Bound Delay in Networked Control Systems". *17th International Symposium on Mathematical Theory of Networks and Systems*, pp. 364-369, 2006.

[79] P. Ogren, E. Fiorelli, and N. E. Leonard. "Cooperative Control of Mobile Sensor Networks: Adaptive Gradient Climbing in a Distributed Environment". *IEEE Transaction for Automatic Control*, vol.49, pp. 1292-1302, 2004.

[80] C. Meng, T. Wang, W. Chou, S. Luan, Y. Zhang, and Z. Tian. "Remote Surgery Case: Robot-assisted Tele Neuro-Surgery". *IEEE International Conference on Robotics and Automation*, vol.1, pp. 819-823, 2004.

[81] J. P. Hespanha, M. L. McLaughlin, and G. Sukhatme. "Haptic Collaboration over the Internet". *Proc. 5th Phantom Users Group Workshop*, 2000.

[82] K. Hikichi, H. Morino, I. Arimoto, K. Sezaki, and Y. Yasuda. "The Evaluation of Delay Jitters for Haptics Collaboration over the Internet". *Proc. IEEE Global Telecomm. Conference*, vol.2, pp. 1492-1496, 2002.

[83] S. Shirmohammadi and N. H. Woo. "Evaluating Decorators for Haptic Collaboration over the Internet". *Proc. 3rd IEEE International Workshop Haptic, Audio and Visual Env. Applic.*, pp. 105-109, 2004.

[84] P. Seiler and R. Sengupta. "Analysis of communication losses in vehicle control problems". *Proc. American Control Conference*, vol.2, pp. 1491-1496, 2001.

[85] P. Seiler and R. Sengupta, "An $\mathcal{H}_\infty$ Approach to Networked Control", *IEEE Trans. Automatic Control*, vol. 50, no. 3, pp. 356-364, March 2005.

[86] R. M. Murray, K. J. Astrorm, S. P. Boyd, R. W. Brockett, and G. Stein. "Control in an information rich world". *IEEE Control System Magazine*, vol.23, pp. 20-33, 2003.

[87] J. H. Braslavsky, R. H. Middleton, and J. S. Freudenberg. "Feedback stabilization over signal-to-noise ratio constrained channels", *Proc. American Control Conference*, pp. 4903-4908, 2004.

[88] N. Elia and S. K. Mitter. "Stabilization of Linear Systems with Limited Information". *IEEE Trans. Automatic. Control*, vol. 46, pp. 1384-1400, 2001.

[89] J. P. Hespanha, A. Ortega, and L. Vasudevan. "Towards the Control of Linear Systems with Minimum Bit-Rate". *Proc. Int. Symposium on Mathematical Theory of Networks and Systems*, 2002.

[90] G. N. Nair and R. J. Evans. "Exponential Stabilizability of finite-dimensional linear systems with limited data rates". *Automatica*, vol.39, pp. 585-593, 2003.

[91] S. Tatikonda and S. Mitter. "Control under communication constraints". *IEEE Trans. Automatic Control*, vol.49, pp. 1056-1068, 2004.

[92] W. S. Wong and R. W. Brockett. "Systems with finite communication bandwidth constraints-II: Stabilization with limited information feedback". *IEEE Trans. Automatic Control*, vol.44, pp. 150–159, 1999.

[93] D. Liberzon and J. P. Hespanha. "Stabilization of nonlinear systems with limited information feedback". *IEEE Transaction on Automatic Control*, vol.50, pp. 910–915, 2005.

[94] G. N. Nair, R. J. Evans, I. M. Y. Mareels, and W. Moran. "Topological feedback entropy and nonlinear stabilization". *IEEE Transaction on Automatic Control*, vol.49, pp. 1585-1597, 2004.

[95] C. M. Krishna and Y. H. Lee. "Special issue on real-time systems". *Proc. IEEE*, vol.91, pp. 983–985, 2003.

[96] M. Cloosterman, N. Wouw, W. Heemels and H. Nijmeijer. "Stabilization of Networked Control Systems with Large Delays and Packet Dropouts". *Proc. American Control Conference*, pp. 4991–4995, 2008.

[97] I. Dumitrache and I. Necoara. "Distributed Control in Networked Systems". Control Engineering and Applied Informatics, vol.11, pp. 3–5, 2009.

[98] B. Frankovic and I. Budinsk. "Creation Of Intelligent Distributed Control System Based on MultiAgent Technology". *Journal of Electrical Engineering*, vol.60, pp. 29-33, 2009.

[99] L. Sheng and Y. Pan. "Distributed control for consensus of networked multi-agent robotic Systems with time delays". *Int. J. Information and Systems Science*, vol.5, pp. 161–178, 2009.

[100] P. Bohner. "A Multi-agent Approach to Distributed Control for Task-level Programs for Cooperating Manipulators". *Proc. International Conference on Intelligent Robots and Systems*, vol.1, pp. 1088–1094, 1994.

[101] Dunbar, W. B. and R. M. Murray. Model Predictive Control of Coordinated Multi-Vehicle Formations. *Proc. Conference on Decision and Control*, pp. 4631–4636, 2002.

[102] Dunbar, W. B. and R. M. Murray. "Distributed receding horizon control for multi-vehicle formation stabilization". *Automatica*, vol.42, pp. 549–558, 2006.

[103] H. Yamaguchi. "A cooperative hunting behavior by mobile robot troops". *Int. J. Robotics Research*, vol.18, pp. 931–940, 2002.

[104] Curtin, T. B., J. G. Bellingham, J. Catipovic and D. Webb. "Autonomous Oceanographic Sampling Networks". Oceanography, pp. 86–94, 1993.

[105] Smith, T. R., H. Hmann and N. E. Leonard. "Orientation control of multiple underwater vehicles". *In. Proc. of 40th Conference on Decision and Control*, pp. 4598–4603, 2001.

[106] L. Buzogany, M. Pachter and J. Azzo. "Automated control of aircraft in formation height". *In. Proc. AIAA Conference Guidance, Navigation and Control*, pp. 1349–1370, 1993.

[107] J. Wolfe, D. Chichka and J. Speyer. "Decentralized controllers for unmanned aerial vehicle formation height". *In. Proc. AIAA Conference Guidance, Navigation and Control*, pp. 3833–3851, 1996.

[108] A. Willig, K. Matheus, and A. Wolisz. "Wireless Technology in Industrial Networks". *Proc IEEE*, vol.93, pp. 1130–1151, 2005.

[109] M. S. Hasan, H. Yu, A. Carrington. Overview of Wireless Networked Control Systems over Mobile Ad-hoc Network. *Proc. 14th International Conference on Automation and Computing*, 2008.

[110] F. D. Pellegrini, D. Miorandi, S. Vitturi, and A. Zanella. "On the Use of Wireless Networks at Low Level of Factory Automation Systems". *IEEE Transactions on Industrial Informatics*, vol.2, pp. 129–143, 2006.

[111] I. F. Akyildiz and I. H. Kasimoglu. "Wireless sensor and actor networks: research challenges". Ad Hoc Networks Journal (Elsevier), vol.2, pp. 351–367, 2004.

[112] C. Perkins and E.M. Royer. Ad-Hoc On Demand Distance Vector Routing in WMCSA, 1999.

[113] D. Johnson and D. A. Maltz. "Dynamic Source Routing in Ad Hoc Wireless Networks. Mobile Computing". pp. 153–181.

[114] Y. Zhu and P. R. Pagilla. "Decentralized Output Feedback Control of a Class of Large-Scale Interconnected Systems". *IMA J. Mathematical Control and Information*, vol.24, pp. 57–69, 2007.

[115] A. S. Titli and N. B. Braiek. "Decentralized Observer based Guaranteed Cost Control for Nonlinear Interconnected Systems". *International Journal of Control and Automation*, vol.2, pp. 29–46, 2006.

[116] Y. Guo, D.J. Hill and Y. Wang. "Nonlinear Decentralized Control of Large-Scale Power Systems". *Automatica*, vol.36, pp. 1275–1289, 2000.

[117] D. Siljak and A. Zecevic. "Control of Large-Scale Systems: Beyond Decentralized Feedback ". *Annual Reviews on Control*, vol.29 , pp. 169–179, 2005.

[118] D.D. Siljak and D.M. Stipanovic. "Robust stabilization of Nonlinear Systems: The LMI Approach". Mathematical Problems in Engineering, vol.6, pp. 461–493, 2000.

[119] X. G. Yan, C. Edwards and S.K. Spurgeon. "Decentralized Robust Sliding Mode Control for a Class of Nonlinear Interconnected Systems by Static Output Feedback". *Automatica*, vol.40, pp. 113–120, 2004.

[120] S. Elloumi and N. Braiek, "Robust Decentralized Control for Multi-Machine Power System: The LMI approach", *Proc. Int. Conference on System Man and Cybernetics*, 2002.

[121] K.S. Narenda and N.O. Oleng. "Exact output tracking in decentralized adaptive control systems". *IEEE Transactions on Automatic Control*, vol.47, pp. 390–395, 2002.

[122] T. Alpcan, L. Pavel and N. Stefanovic. "A Control Theoretic Approach to Non-cooperative Game Design". *Proc. 48th IEEE Conference on Decision and Control*, pp. 8575–8580, 2009.

[123] Y. Tipsuwan, M. Chow. "Control methodologies in networked control systems". Control Engineering Practice, vol.11, pp. 1099-1111, 2003.

[124] H. Yin, U. Shanbhag and P. Mehta ."Nash Equilibrium Problems with Congestion Costs and Shared Constraints". *IEEE Transactions on Automatic Control*, vol.56, pp. 1702–1708, 2011.

[125] T. Alpcan, L. Pavel and N. Stefanovic. "Control Theoretic Approach to Non-cooperative Game Design". *Proc. of the 48th IEEE Conference on Decision and Control*, pp. 8575–8580 , 2009.

[126] A. Gattami. "Distributed Stochastic Control: A Team Theoretic Approach". *Proc. of the 17th International Symposium on Mathmatical Theory of Networks and Systems*, 2006.

[127] A. Rantzer. "Dynamic Dual Decomposition for Distributed Control". *American Control Conference* 2009.

[128] S. Mittal, A. S. Siddiqui. "Networked Control System: Survey and Directions". *Journal of Engineering Research and Studies*, vol.1, pp. 35–50, 2010.

[129] E. Witrant, P. Park, M. Johansson, C. Fischione and K. Johansson. "Predictive control over wireless multi-hop networks". *IEEE International Conference Control Applications*, pp. 1037–1042, 2007.

[130] M. Drew, X. Liu, A. Goldsmith, and K. Hedrick. "Networked Control System Design over a Wireless LAN". *Proc. 44th IEEE Conference on Decision and Control, and the European Control Conference*, pp. 6704–6709, 2005.

[131] D. Yue, Q. Han, and C. Peng. "State Feedback Controller Design of Networked Control Systems", *IEEE Trans. Circuits and SystemsII: Express Briefs*, vol. 51, no. 11, November 2004.

[132] P. Voulgarisa, G. Bianchinib and B. Bamiehc, "Optimal H_2 Controllers for Spatially Invariant Systems with Delayed Communication Requirements", *Systems and Control Letters*, vol. 50, pp. 347–361, 2003.

[133] M. Long and W. Chu, "Robust Stabilization of Discrete-Time Systems with Time-Varying Delays", *Proc. American Control Conference*, June 2005.

[134] M. S. Mahmoud, "Switched Discrete-Time with Time-Varying Delays: A Generalized H_2 Approach", *Computers and Mathematics with Applications*, vol.57, no.1, pp. 79–95, 2009.

[135] G. Chen and F. L. Lewis, "Distributed Adaptive Tracking Control for Synchronization of Unknown Networked Lagrangian Systems", *IEEE Trans. Systems, Man, and Cybernetics: Part B*, vol. 41, no. 3, 2011, pp. 805–816.

[136] P. D. Wilde and G. Briscoe, "Stability of Evolving Multiagent Systems", *IEEE Trans. Systems, Man, and Cybernetics: Part B*, vol. 41, no. 4, pp. 1149–1157, 2011.

[137] Z. G. Wu, P. Shi, H. Su, and J. Chu, "Delay-Dependent Stability Analysis for Switched Neural Networks With Time-Varying Delay", *IEEE Trans. Systems, Man, and Cybernetics: Part B*, vol. 41, no. 6, pp. 1522–1530, 2011.

[138] H. Zhang, C. Dang and J. Zhang, "Decentralized Fuzzy H_∞ Filtering for Nonlinear Interconnected Systems With Multiple Time Delays", *IEEE Trans. Systems, Man, and Cybernetics: Part B*, vol. 40, no. 4, pp. 1197–1203, 2010.

[139] H. Zhang, D. Yang, and T. Chai, "Guaranteed Cost Networked Control for TS Fuzzy Systems With Time Delays", *IEEE Trans. Systems, Man, and Cybernetics: Part C*, vol. 34, no. 2, pp. 160–172, 2007.

[140] Y. Zhao, C. Zhang, and H. J. Gao, "A New Approach to Guaranteed Cost Control of T-S Fuzzy Dynamic Systems With Interval Parameter Uncertainties", *IEEE Trans. Systems, Man, and Cybernetics: Part B*, vol. 39, no. 6, pp. 1516–1527, 2009.

[141] J. Schendel, M. Donkers, W. Heemels and N. Wouw. "On Dropout Modeling for Stability Analysis of Networked Control Systems", *Proc. American Control Conference*, pp. 555–561, 2010.

[142] J. Singh, B. Patra and S. Singh, "An Algorithm to Reduce the Time Complexity of Earliest Deadline First Scheduling Algorithm in Real-Time System". *Int. J. of Advanced Computer Science and Applications*, vol. 2, pp. 31–37, 2011.

[143] L. LI and F. Wang, "Control and Communication Synthesis in Networked Control Systems", *Int. J. Intelligent Control and Systems*, vol. 13, pp. 81–88, 2008.

[144] D. Nesic and D. Liberzon, "A Unified Approach to Controller Design for Systems with Quantization and Time Scheduling", *Proc. 46th IEEE Conference on Decision and Control*, pp. 3939–3944, 2007.

[145] A. Matveev and V. Savkin, "Decentralized Stabilization of Networked Systems under Data-Rate Constraints", *Proc. of the 17th World Congress*, pp. 8034–8038, 2008.

[146] S. Yuksel and T. Basar, "Communication Constraints for Decentralized Stabilizability with Time-Invariant Policies", *IEEE Trans. Automatic Control*, vol. 52, pp. 1060–1066, 2007.

[147] D. Dacic and D Nesic, "Simultaneous Controller and Protocol Design for Networked Control", *Proc. IEEE Conf. on Decision and Control*, pp. 508–514, 2006.

[148] Y. Xia, J. Shang, J. Chen, and G. Liu. "Networked Data Fusion with Packet Losses and variable Delays". *IEEE Trans. Systems, Man, and Cybernetics: Part B*, vol. 39, no. 5, 2009, pp. 1107–1120.

[149] J. K. Yook, D. M. Tilbury, N. R. Soparkar. "Trading Computation for Bandwidth:Reducing Communication in Distributed Control Systems Using State Estimators". *IEEE Transactions on Control System Technology*, July 30, 2000.

[150] M. Hompel, S. Libert. "Distributed Control Nodes for Material Flow System Controls on the Example of Unit Load Conveyor and Sorter Facilitles". Logisitcs Journal, ISSN 1860-7977, 2006.

[151] N. Motee1 and A. Jadbabaie. "Approximation Methods and Spatial Interpolation in Distributed Control Systems". *American Control Conference*, pp. 860–865, 2009.

[152] M. Postol. "Real-Time Communication for Large Scale Distributed Control Systems". Proceedings of the International Multiconference on Computer Science and Information Technology pp. 859-869, 2007.

[153] A. Antunes, A.Manuel Mota. "Control Performance of a Real-time Adaptive Distributed Control System under Jitter Conditions". Control Journal, 2004.

[154] J. K. Yooka, D. M. Tilburya and N. R. Soparkarb. "A design methodology for distributed control systems to optimize performance in the presence of time delays". *International Journal of Control*, vol. 74, no. 1, pp. 58–76, 2001.

[155] D. Nesic and D. Liberzon. "A unified approach to controller design for systems with quantization and time scheduling". *Proc. of the 46th IEEE Conference on Decision and Control*, pp. 3393 – 3944, December , 2007.

[156] E.I. Silva, D.E. Quevedo, M.S. Derpich and G.C. Goodwin. "Design of Feedback Quantizers for Networked Control Systems". Technical report, School of Electrical Engineering and Computer Science, The University of Newcastle, Australia, 2008.

[157] Luis Montestruque, Panos J. Antsaklis, Quantization in Model Based Networked Control Systems, *Proc. of the 16th IFAC World Congress*, Prague, Czech Republic, July 4-8, 2005.

[158] Y. Ishido, D. E. Quevedo, K. Takaba. "Stability Analysis of Networked Control Systems Subjected to Packet-dropouts and Finite Level Quantization". Systems and Control Letters Vol. 60, Issue 5, pp. 325–332, 2011.

[159] H. Fang, P. Antsaklis. "Convergence Rate of Quantization Error in Networked Control Systems". *14th Mediterranean Conference on Control and Automation* (MED06), Universit Politecnica delle Marche, Ancona, Italy, June 28-30, 2006.

[160] R. Scattolini. "Architectures for distributed and hierarchical Model Predictive Control A review". *Journal of Process Control*, pp. 723 - 731, 2009.

[161] F. Al-Nasser and H. Rowaihy, "Simulation of Dynamic Traffic Control System Based on Wireless Sensor Network", *IEEE Symposium on Computers and Informatics(ISCI 2011)*, pp. 302–307, 2011.

[162] F. Al-Nasser and M. Mahmoud. Chapter title: Wireless Sensors Network Applications: A Decentralized Approach for Traffic Control and Management, Book title: Wireless Sensor Networks - Technology and Applications, InTech, ISBN $978 - 953 - 51 - 0676 - 0$, Published July 18, 2012.

[163] K. Tan, M. Khalid and R. Yusof. "Intelligent Traffic Lights Control By Fuzzy Logic" , *Malaysian Journal of Computer Science*, Vol. 9 No. 2, pp. 29–35, December 1996,

[164] L. K. Chien, "A Real-time Dynamic Traffic. Control System Based on Wireless Sensor Network". *International Conference on Parallel Processing Workshops - ICPPW*, 2005.

[165] S. Mehta. "Fuzzy Control System For Controlling Traffic Lights". *Proc. the International Multi Conference of Engineers and Computer Scientists (IMECS)*, vol. 1, 19-21 March, 2008.

[166] WEN Y, PAN J. L. and LE J. F; "Survey On Application Of Wireless Sensor Networks For Traffic Monitoring", *International Conference on Transportation Engineering - ICTE*, 2007.

[167] M. Tubaishat, Y. Shang, H. Shi. "Adaptive Traffic Light Control with Wireless Sensor Networks" , *4th Consumer Communications and Networking Conference - CCNC*, 2007.

[168] A. Albagul, M. Hrairi, Wahyudi and M. F. Hidayathullah; "Design and Development of Sensor Based Traffic Light System", *American Journal of Applied Sciences*, pp. 1745–1749, 2006.

[169] W. Marco, V. Jelle, V. Jilles and Arne K. "Intelligent Traffic Light Control" . Institute of Information and Computing Sciences, Utrecht University, Technical Report UU-CS-2004-029 , July 9, 2004.

[170] P. Lotito, E. Mancinelli and J. Quadrat. "A Min-Plus Derivation of the Fundamental Car-Traffic Law", *IEEE-AC V.50*, N.5, pp. 699–705, 2005.

[171] D. Seborg, D. Mellichamp, T. Edgar, F. Doyle, III. Process Dynamics and Control. John Wiley and Sons, Apr 12, 2010.

[172] E. Mancinelli ,G. Cohen ,S. Gaubert ,J. Quadrat and Rofman J. On Traffic Light Control of Regular Towns. *INRIA Report* Sept. 2001.

[173] A. Perkusich, L. Arajo ,L. Coelho ,L. Gorgnio ,L. Ribeiro and L. Lemos. "Design and Animation of Coloured Petri Nets Models for Traffic Signals". *Second Workshop and Tutorial on Practical Use of Co loured Petri Nets and Design CPN*, 1999.

[174] Lefei L. "Traffic signal priority/preemption control with colored Petri nets". *Proc. IEEE Intelligent Transportation Systems*, pp. 694–699, 2005.

[175] A. Ganiyu ,O. Olabiyisi ,O. Omidiora ,O. Okediran and O. Alo. "Modelling and validation of a multi-phase traffic light controlled cross-type intersection using timed coloured petri nets". *American Journal of Scientific And Industrial Research*,vol 2(5), pp. 807–819, 2011.

[176] M. Martn ,R. Mnica and F. Alberto. "A Propose-and-revise System for Real-time Traffic Management". *European Symposium on Intelligent Techniques*, pp. 62–70, Sep. 2000, Aachen, Germany.

[177] H. Josefa, C. Jose and M. Martin. "Real-time Traffic Management Through Knowledge-based Models: The TRYS Approach". Tutorial on Intelligent Traffic Management Models, Helsinki 1999, Finnland.

[178] Z. Nan, L. Pei-Wei and C. Gang-Len ,R. Saed. "An Integrated Knowledge-Base/Simulation System for Realtime Incident Management". Department of Civil and Environmental Engineering, University of Maryland at College Park, 2003.

[179] K. Michal,S. Zden and C. Anton. "Balancing the Waiting Times in a Simple Traffic Intersection Model".*The 11th IFAC Symposium on Control in Transportation Systems*, 2006.

[180] W. Van and T. Vandaele. "Empirical validiation of a queueing approach to uninterrupted traffic flows", *UFSIA research paper* , 2002.

[181] V. Nico, W. Tom and A. Aviel. "Queueing Based Traffic Flow Model". *Transportation Research-D: Transport and environment*, vol. 5 no. 2, pp. 121–135, January 2000.

[182] A. Hegyi, B. De Schutter, R. Babu ,S. Hoogendoorn,H. Zuylen and H.Schuurman. "A fuzzy decision support system for traffic control centers". *Proc. the TRAIL 6th Annual Congress Transport, Infrastructure and Logistics*, Part 2, The Hague/Scheveningen, The Netherlands, 10, Dec. 2000.

[183] R. Narayanan, ,R. Udayakumar, K. Kumar and L.Subbaraj. "Quantification of congestion using Fuzzy Logic and Network Analysis using GIS". *Proc. of Map India Conference*, New Delhi, India: Jan. 28-31, 2003.

[184] G. Jiang,J. Wang, ,X. Zhang and L.Gang, "The study on the application of Fuzzy Clustering Analysis in the dynamic identification of road traffic state". *IEEE 6th International Conference on Intelligent Transportation Systems*, Shanghai, Oct. 12-15, 2003.

[185] I. Are, C. Liu T., Urbanik A.G. and Kohls. "Reinforcement learning-based multi-agent system for network traffic signal control". *IET Intelligent Transport Systems*, vol. 4, no. 2, pp. 128-135, 2010.

[186] C. Silva ,D. Oliveira ,C. Bazzan and W. Basso. "Adaptive traffic control with reinforcement learning". *Proc. the 4th Workshop on Agents in Traffic and Transportation (AAMAS)*, pp. 80-86, May 2006.

[187] B. Abdulhai. "Reinforcement Learning for True Adaptive Traffic Signal Control". *ASCE Journal of Transportation Engineering*, 129(3): pp. 278–285, 2003.

[188] B. Bakker, M. Steingrover, R. Schouten, E. Nijhuis and L. Kester. "Cooperative multi-agent reinforcement learning of traffic lights". *Proc. of the Workshop on Cooperative Multi-Agent Learning, European Conference on Machine Learning (ECML)*, 2005.

[189] S.Leena ,T. Sudhanshu and A. Himakshi. "Time Optimization for Traffic Signal Control Using Genetic Algorithm", *International Journal of Recent Trends in Engineering*, Vol 2, No. 2, November 2009.

[190] V. Hirankitti, J. Krohkaew and C. Hogger. "A Multi-Agent Approach for Intelligent Traffic-Light Control", *Proc. of the World Congress on Engineering - WCE*, vol.1, July , London, U.K. , 2007.

[191] Z. Josefa , O. Sascha ,G. Ana. "Multiagent Architectures for Intelligent Traffic Management Systems". *Transportation Research Part C 10*, pp. 473-506, 2002.

[192] Y. ZHAO-SHENG, C. XIN, T. YANG-SHAN and JIAN P. "Intelligent Cooperation Control of Urban Traffic Networks". *Proc. the 4th International Conference on Machine Learning and Cybernetics*, Guangzhou, 18-21 August 2005.

[193] C. Kuei-Hsiang ,L. Ren-Hao and W. Meng-Hui. "An Intelligent Traffic Light Control Based on Extension Neural Network." *In Knowledge-Based Intelligent Information and Engineering Systems*, 2009.

[194] A. de la Escalera, J. M. Armingol and M. Mata. "Traffic sign recognition and analysis for intelligent vehicles." *Image and Vision Computing*, 21, pp. 247–258, 2003.

[195] A. Serrano ,C. Cristina ,R. Licesio, M. Raquel and C. Enrique. "Computer Vision Application: Real Time Smart Traffic Light." *In Computer Aided Systems Theory EUROCAST*, 2005.

[196] P. Shepherd. "A Review of Traffic Signal Control", *Working paper 349, Institute of transport Studies*, University of Leeds, 1992.

[197] M. Pitu , W. Fei-Yue. "RHODES to Intelligent Transportation Systems", *IEEE Intelligent Systems Journal*, vol. 20, No. 1, January 2005.

[198] A. Montasir,C. Hassan, C. Nadeem and J. Youn. "Distributed Architecture and Algorithm For Robust Real-Time Progression Evaluation and Improvement", *Technical Report 0-4729-2, Project Number 0-4729*, Texas University, 2006.

[199] C. Liu, H. Tao and Tabacek. "Empirical observations of dynamic dilemma zones at signalized intersections", *Proc. the 86th meeting of the Transportation Research Board*, July 2006.

[200] F. Martin and S. Radiocomun, "Vehicular Traffic Surveillance and Road Lane Detection Using Radar Interferometry", *IEEE Trans. Vehicular Technology*, vol. 61, pp. 959–970, March 2012.

[201] S. Jiaotong and A. Wang, "Study of Self-organizing Control of Traffic Signals in an Urban Network Based on Cellular Automata".*IEEE Trans. Vehicular Technology*, vol. 54, pp. 744–748, March 2005.

[202] J. Xu, Z. Hou, H. Zhong. "Freeway Traffic Control Using Iterative Learning Control-Based Ramp Metering and Speed Signaling", *IEEE Trans. Vehicular Technology*, vol. 56, pp. 466–477, March 2007.

[203] Y. Liu, G. Chang and J. Yu. "An Integrated Control Model for Freeway Corridor Under Non-recurrent Congestion", *IEEE Trans. Vehicular Technology*, Vol. 60, pp. 1404–1418, May 2011.

[204] S. Eben Li , H. Peng , K. Li and J. Wang, "Minimum Fuel Control Strategy in Automated Car-Following Scenarios", *IEEE Trans. Vehicular Technology*, vol. 61, pp. 998–1007, March 2012.

[205] M. Mazo and P. Tabuada, "Decentralized Event-Triggered Control Over Wireless Sensor/Actuator Networks", *IEEE Trans. Automatic Control*, vol. 56, pp. 2456–2461, Oct. 2011.

[206] A. Gning, L. Mihaylova and R. K. Boel, "Interval Macroscopic Models for Traffic Networks", *IEEE Trans. Intelligent Transportation Systems*, vol. 12 , pp. 523–536, June 2011.

[207] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos, "Store-and-forward based methods for the signal control problem in large-scale congested urban road networks". Transportation Research Part C-Emerging Technologies, vol. 17, pp. 163-174, 2009.

[208] C. Joseph, V. George, P. Venkateswaran. "Discrete Controller Design for A Hybrid Three Tank System". International Journal on Advanced Science Engineering Information Techonology, vol. 2, no. 1 , pp. 40–44, 2012.

[209] Federal Highway Administration Research and techonology. Traffic Detector Handbook: Third EditionVolume II. Publication Number: FHWA-HRT-06-139, October 2006 .

[210] Federal Highway Administration Research and techonology. Traffic Control Systems Handbook: Chapter 3. Control Concepts - Urban And Suburban Streets.

Vitae

- Name: Faisal Ahmed Al-Nasser
- Nationality: Syrian
- Date of Birth: 02/02/1980
- Email: *g199785010@kfupm.edu.sa*
- Permanent Address: Saudi Arabia , Industrial Jubail City

  Faisal Ahmed Al-Nasser was born in Syria 1980. He finished a double majors in engineering at King Fahd University of Petroleum and Minerals, (KFUPM), B. Sc. in Systems Engineering with Automation and Control and the 2nd BSc in Computer Engineering, both finished in 2003. Then MSc was completed in Systems Engineering by 2007 as part - timer student and working at the same time as Senior Process Automation Engineer at Saudi Iron and Steel Company - HADEED. Then, the PhD was completed in the same field and same manner in Systems Engineering. Examples for areas of interest are: Control Networks, Control Systems, Multimedia Communications, Signalized Traffic Intersections Coordination and Control, WLAN, WSN and more. He has several published conference papers and a book chapter.