IVC & ITS 2009

Oleg Gusikhin
Maria Pia Fanti (Eds.)

Intelligent Vehicle Controls & Intelligent Transportation Systems

Proceedings of the
3rd International Workshop on
Intelligent Vehicle Controls & Intelligent
Transportation Systems - IVC & ITS 2009

In conjunction with ICINCO 2009
Milan - Italy, July 2009

INSTICC PRESS

Oleg Gusikhin and
Maria Pia Fanti (Eds.)

Intelligent Vehicle Controls & Intelligent Transportation Systems

Proceedings of the
3rd International Workshop on
Intelligent Vehicle Controls & Intelligent Transportation Systems
Workshop IVC & ITS 2009

In conjunction with ICINCO 2009
Milan, Italy, July 2009

INSTICC PRESS
Portugal

Volume Editors

Oleg Gusikhin
Ford Research & Adv.Engineering
U.S.A.

and

Maria Pia Fanti
Polytechnic of Bari
Italy

3rd International Workshop on
Intelligent Vehicle Controls & Intelligent Transportation Systems
Milan, Italy, July 2009

Printed in Portugal

ISBN: 978-989-674-003-0
Depósito Legal: 294963/09

Foreword

The subject of Intelligent Vehicle Controls & Intelligent Transportation Systems covers a broad interdisciplinary area of the research and development toward next generation mobility solutions. The topics of interest range from computational intelligence methods in vehicle safety applications and autonomous vehicle technologies to new business models based on advancements of transportation information support infrastructure. The goal of the workshop is to bring together representatives from academia, industry and government agencies to exchange ideas on state of the art in intelligent vehicle controls and intelligent transportation systems.

The special interest of this year's workshop is the impact of the advancements in intelligent transportation systems and information technologies on the commercial logistics and supply chain management infrastructure.

Information and Communication Technologies are considered to be the key tools to improve efficiency and safety in transportation systems. In recent years, the European Union has sponsored several projects targeting advancements of different transportation systems, such as CRObiT (Cross Border Information Technology) that addresses railways transportation and MarNIS, an integrated project aiming to establish maritime navigation information structure in European waters.

A number of projects focus specifically on efficient freight transpiration: Freight-wise aims to establish a framework for efficient comodal freight transport on the Norwegian ARKTRANS system; e-Freight is a continuation of Freightwise to promote efficient and simplified solutions in support of cooperation, interoperability and consistency in the European Transport System; the SMARTFREIGHT project (FP7 – ICT Objective 6.1) addresses the efficiency of urban freight transport.

The workshop keynote speaker is Dr. Paolo Paganelli from INSIEL. Dr. Paganelli is the leader of the European project ËURIDICE-European Inter-disciplinary Research on Intelligent Cargo for Efficient, Safe and Environmental-friendly Logistics". EURIDICE is a project seeking to develop an advanced European logistics system around the concept of 'intelligent cargo'.

The workshop includes 14 papers: 12 full papers and 2 posters.

The presentations that have been selected are consistent with the two relevant topics and special interest of the workshop on the freight transportation.

The papers on Intelligent Vehicle Control Systems include adaptive cruise control and vehicle safety applications, control and navigation of unmanned ground and aerial vehicles. The papers on ITS infrastructure address traffic management systems and vehicle communication.

The presentations on freight management are organized into three special sessions and include discussions on dynamic routing of freight vehicles using ITS information; meta-modeling of complex logistics networks and information systems; planning of railway freight transportation; management of transportation of hazardous materials on congested motorways; effect of dispatching policies on the stability and optimality of the logistics networks; decision support for management of flow of goods through the logistic traffic network; and indoor location tracking of forklifts and containers.

We would like to thank all the authors for their contribution to this workshop, members of program committee and reviewers for providing their support, suggestions and comments on the technical directions, and thorough reviews for the papers. In addition, we would like to acknowledge Professor Walter Ukovich for his efforts in organizing special sessions. Finally, we would like to express our gratitude to ICINCO Organization Committee, with special thanks to Professor Joaquim Filipe, Marina Carvalho and Vitor Pedrosa for their help and assistance in the organization of the workshop.

July 2009,

Oleg Gusikhin
Ford Research & Adv. Engineering, U.S.A.

Maria Pia Fanti
Polytechnic of Bari, Italy

Workshop Chairs

Oleg Gusikhin
Ford Research & Adv. Engineering
U.S.A.

and

Maria Pia Fanti
Polytechnic of Bari
Italy

Invited Speakers

Paolo Paganelli, Insiel S.p.A., Italy

Program Committee

Mauro Boccadoro, DIEI - Università di Perugia, Italy
Valentina Boschian, University of Trieste, Italy
Giampaolo Centrone, Università degli Studi di Trieste, Italy
Ratna Babu Chinnam, Wayne State University, U.S.A.
Mark Everson, Ford Research & Advanced Engineering, U.S.A.
Davide Giglio, University of Genova, Italy
T. J. Giuli, Ford Motor Company, U.S.A.
Riad Hammoud, Delphi Corporation, U.S.A.
Erica Klampfl, Ford Research & Advanced Engineering, U.S.A.
Ilya Kolmanovsky, Ford Research and Advanced Engineering, U.S.A.
Anatoli Koulinitch, ArvinMeritor, U.S.A.
Perry MacNeille, Ford Motor Company, U.S.A.
James Miteff, Ford Research & Advanced Engineering, U.S.A.
Alper E. Murat, Wayne State University, U.S.A.
Kwaku Prakah-Asante, Ford, U.S.A.
Venkatesh Prasad, Ford Motor Company, U.S.A.
Danil Prokhorov, Toyota Tech Center, U.S.A.
Christian Ress, Ford Motor Company, Germany
Silvia Siri, University of Genova, Italy
Gabriella Stecco, Università degli Studi di Trieste, Italy

Kacie Theisen, Ford Motor Company, U.S.A.
Walter Ukovich, University of Trieste, Italy
Diana Yanakiev, Ford Motor Co., U.S.A.

Table of Contents

Invited Speakers

Full Papers

Posters

INVITED SPEAKERS

The Intelligent Cargo Concept in the European Project Euridice

Paolo Paganelli

Insiel S.p.A., Italy
EURIDICE Project Coordinator

Extended Abstract

The special interest of this year's workshop on the impact of the advancements in intelligent transportation systems and information technologies on the commercial logistics is noteworthy in line with the key concept of 'intelligent cargo' developed by the European project "EURIDICE-European Inter-disciplinary Research on Intelligent Cargo for Efficient, Safe and Environmental-friendly Logistics".

EURIDICE aims to create the necessary concepts, technological solutions and business models to establish the most advanced information services for freight transportation in Europe. The project is built on the 'intelligent cargo' concept, defined by the following vision statement: "in five years time, most of the goods flowing through European freight corridors will be 'intelligent', i.e.: self-aware, context-aware and connected through a global telecommunication network to support a wide range of information services for logistic operators, industrial users and public authorities."

The problems addressed by EURIDICE are well known to the logistics community, and as such have inspired a large number of initiatives and projects during the last decades, both in industry and in Scientific ant Technological (S/T) research. These initiatives take their inspiration from the complexity of the logistic business, characterized by an high variety of involved actors, complex processes, multiplicity of impacts both on companies and citizens.

Due to important scientific and industrial developments in the last decade the basic technological components supporting the 'intelligent cargo' concept are available, although at different levels of maturity. These include Radio-Frequency Identification technologies (RFID), service oriented architectures (SOA) and interoperability platforms for data interchange and collaboration between business partners, mobile technologies and global positioning systems. Equally important is the availability of standards addressing all the aspects of cargo identification and management, from RFID tag specifications, to identification of individual items (EPC) and shipments (GTIN), to definition of logistic data and processes.

For their very nature technology developments and standardization initiatives aim at removing specific technical barriers to innovation, each representing just a portion, if important, of the whole problem of intelligent monitoring and management of moving goods. For this reason, the EURIDICE project has been launched with the aim to integrate state-of-the-art results and fill existing gaps in cargo-related technologies to deliver solutions based on the intelligent cargo vision.

The full realization of this vision will have a significant impact in terms of diffusion and effectiveness of ICT support to freight transportation, producing relevant benefits for businesses and the society: (i) enhanced and widespread capability to monitor, trace and safely handle moving goods at the required level of detail, from full shipments to individual packages or items; (ii) increased efficiency of transportation networks, by improving synchronization between logistic users, operators and control authorities; (iii) improved sustainability of logistic systems, by reducing their impact on local communities in terms of traffic congestion and pollution.

To achieve these benefits, the EURIDICE project intends to introduce a paradigm shift on how moving goods are currently handled by ICT systems. Currently freight information is "pushed" by back-office systems, and system-to-system communication is the only interoperability paradigm. As a consequence, the information flows are often misaligned with the actual flow of goods. The EURIDICE approach is to connect cargo objects (items, containers, vehicles and infrastructure) with each other, to provide intelligent services for logistics stakeholders anywhere needed along the route. The main changes entailed by the shift to Intelligent Cargo are the following:

- From "organization-to-organization", where information exchanges happen between organizations' centralized systems, to "thing-to-thing", where information is exchanged through independent connection of cargo objects, data processing is as much as possible decentralized, and backoffice systems are involved only if needed.
- From predefined links, either one-to-one or via interchange "hubs", to any-to-any communication and data interchange, based on DNS-like system for cargo objects and related services, globally shared semantics, on demand configuration of communication resources.
- From centralized decisions support, based on data consolidation from back-office systems and top-down monitoring, revision and communication, to event-triggered, decentralized decisions support based on automated event detection, context determination and bottom-up exception resolution.

In this way it is possible to create a new paradigm for freight information services that covers not only basic but also advanced capabilities of 'Intelligent Cargo':

- Cargo capable of autonomous decisions (intelligent agent)
- Cargo capable to start a service (independent behavior)
- Cargo capable to monitor and register its status
- Cargo capable to grant access to services (authorization, ETA, data read/write, ..)
- Cargo capable to detect its context (location, user, infrastructure, ..)
- Cargo capable to identify itself.

The list is sorted by decreasing level of 'intelligence'. Clearly from EURIDICE Point of view, the 'intelligent cargo' is not defined essentially by the application of artificial intelligence or other advanced technologies. It is rather a paradigm for cargo interaction that may include those technologies, but may also be implemented with conventional technologies deployed following a different architectural approach.

Different intelligent cargo capabilities require different implementation models, e.g., basic capabilities should be available as public domain services for all the intelligent cargo users and specialized capabilities should be developed for specific purposes by individual users or groups of users to fulfill specific application requirements.

FULL PAPERS

Co-operative Traffic Light: Applications for Driver Information and Assistance

Franziska Wolf, Stefan Libbe and Andreas Herrmann

Institut f. Automation und Kommunikation
Werner-Heisenberg Str. 1, 39106 Magdeburg, Germany
{Franziska.Wolf, Stefan.Libbe, Andreas.Herrmann}@ifak.eu
http://www.ifak.eu

Abstract. In this paper a new approach towards co-operative intersection management systems based on traffic flow detection and analysis shall be proposed which uses on the one hand standard in-vehicle equipment and on the other hand standard traffic light control systems. The main communication concept will be placed into a low-cost modular unit which connects the systems of the vehicles and the traffic light control systems in order to enable traffic data information exchanged such as traffic light switching information for the individual traffic and speed recommendation for a co-operative traffic light management.

1 Introduction

The amount of traffic has strongly increased in recent years. In many places the infrastructures are not able to react to the traffic increase efficiently, whether by road construction or rebuilding of infrastructure. The continuous increase in traffic and environmental problems as well as the demographic change are a challenge in many regions of Germany and Europe. The expansion and reconstruction of the traffic infrastructure are still advanced in many countries, however this is mainly concentrated on maintenance and repair in German regions and conurbations. With the use of existing infrastructures, an innovative, telematic based traffic management offers new options.

1.1 Current Traffic Light Control Systems

The transport infrastructure (road, rail, water and air) provides mobility to our society today. Improvements to this infrastructure enhance this mobility. Unfortunately the communication infrastructure has not been able to keep the pace of these developments. Recent developments in digital networks, however, allow the communication infrastructure to catch up.

Nowadays the basic principals of traffic management are based on the measurement and control of traffic flows. One basic mechanism for this is the usage of traffic light control systems. This is especially true for inner-urban areas where signal control substantially determines the traffic and mobility management. Because the traffic

light controls are very important for the traffic management of today, their impact to traffic flows can be assumed as very denotative.

The traffic light control systems are based on signal programs which are in general adapted to the prevailing traffic situations. In the planning phases of the traffic management development and in planned intervals, the expected traffic situations are analyzed using traffic flow measurement and various analysis mechanisms such as simulation 1 or traffic models 234. The cycles of the signal programs of the traffic light controls are chosen in order to cover the varying load levels as they might occur during daytimes, different days of a week or at special times.

These systems shall generally increase traffic safety and improve the traffic flow quality. The following criteria 5 can be proposed in order to define the traffic light control programs adapting to the traffic situations:

- traffic volume
- interrelations between the traffic volumes
- degree of occupancy
- speeds

The signal programs are mainly based on two coordination principles which are the time dependent traffic light control and traffic-actuated traffic light control. Normally the traffic light control programs vary over times of days and weeks between programs following the time and traffic dependant strategies.

The time dependent traffic light controls are especially useful for forecasted high-load periods of days of weeks. They offer the advantages that these traffic flows can be diverted effectively, especially for streets of known heavy traffic. A strategy which can be particularly used here is the progressive signal system such as Green Wave where the signals of multiple light-signal systems are switched consecutively.

Traffic-actuated traffic light control programs offer the strategy to adapt the switching to the local traffic which is detected around the intersection. The road networks are full of detectors, but these detectors are specialised for local traffic light controls. For this reason they are mostly located near intersections which are controlled. For traveller information pre-trip and on tour, traffic parameters not usually measured for local traffic light control are of more interest, for example the speed profile along stretches. Furthermore these mechanisms offer short term traffic adoption but are for now limited to light traffic situations as they are assumed beforehand. For instance in situations when there is a sudden rise of traffic volume, such as during detours or after public events, the traffic flow can often not be managed satisfyingly by strategies of traffic management due to a lack of feedback coming from the local traffic detectors on the one hand and a limited foresight of these detectors concerning the upcoming traffic on the other hand. Several strategies to solve these problems have been proposed recently, one of them is to use the local vehicle movement of data in order to measure tailbacks in the inflow of traffic light controlled intersections 67. The new concept that shall be proposed here aims at a co-operative approach where the collaboration of vehicles together with the infrastructure tend to achieve a better traffic organization for both sides.

1.2 Co-operative Concept

Currently a high fraction of vehicles in the individual traffic is equipped with up-to date ITS such as navigation systems or PDAs enabling well known HMI, GPS and WLAN or GSM communication modules. These in-vehicle systems are already able to measure and calculate the vehicles own Floating Car Data (FCD). The proposed new approach of traffic control aims to make the already calculated parts of traffic flow data of the traffic participant available to the road authorities in order to improve the management of traffic flows by traffic light control systems.

On the one hand standard in-vehicle equipment shall be used along with standard traffic light control systems. The main communication concept will be placed into a low-cost modular unit which connects the systems of the vehicles and the traffic light control systems in order to enable traffic information exchanged, such as traffic light switching information for the individual traffic and traffic flow information for a co-operative traffic light control management.

On the basis of transferred information between vehicle and traffic lights, the local control can be optimised by supporting the driver on the one hand and on the other by making the traffic light control aware of the approaching traffic flows. The interaction between vehicle and infrastructure can help to improve both traffic flows and advanced driver assistance systems. The research proposed here aims to gather the requirements of co-operative traffic light control systems 89 and to characterise the results of the first developed prototype in order to prepare an industrial development of a practicable co-operative technology.

2 Requirements for a Co-operative Traffic Light Control

The main aim is the combination of the advantages of the global and the local intersection management. It shall lead to a management of the traffic flows based on both local feedback of detections together with progressive signal systems (e.g. Green wave) over a wider area of the road network. This enables advantages for both: the road authorities because of reduction of the environmental pollution and noise, and enhancement of the public road traffic. Furthermore it brings advantages for individual traffic participants because their travel times and fuel consumption can be reduced and therefore the acceptance towards co-operative traffic strategies can be enhanced. In order to achieve such an advancement of intersection management, different requirements must be fulfilled for the traffic light control systems and for the drivers.

The main requirement for the traffic light control systems at intersections is:

- Optimisation of the traffic light phases to the traffic flows

Therefore the traffic light systems need information on the traffic flows which consist of:

  - Current speed of vehicles
  - Direction of vehicles
  - Current position of vehicles

The main requirement towards the traffic participants is:

- Optimisation of the vehicle speed towards the traffic light phases (progressive speed)
- Reduction of stops

In order to encourage the participants of the traffic to drive co-operatively, the following information can be offered:

  - Announcement of traffic light switching to the drivers
  - Information about optimized speed in local sections of the intersections

This information can also enhance the positive side effects of co-operative traffic management systems towards the reduction of environmental pollution because the drivers of the vehicle groups can be animated to an optimised driving speed and an early off-switching of engines when waiting for the next green phase.

2.1 Requirements towards Investment and Safety Reasons

Apart from the function of the co-operative system itself, some more requirements have to be regarded in order to realize an effective co-operative intersection management especially regarding acceptance:

- The equipment used should be cost efficient for the drivers and the local authorities

The efficiency of a co-operative system depends on the penetration rate (significant amount of equipped vehicles) This can only be achieved by using systems that are already part of most cars, such as navigation systems, PDAs or mobile phones. For the public authorities, the requirement results in the original usage or a low-cost upgrade of the control systems in use.

The aim is to enable a maximum benefit with minimum costs.

- Installation in traffic light control systems, regarding safety and security issues, requires standardized interfaces and long lasting and modular components

In order to avoid safety and security failures, a modular device shall be developed which is connected to the traffic light control systems via standardized interfaces, like they have been proposed in 9. The modular and standardized approach offers one more positive side effect: the device can be applied and removed wherever and as long as it is needed at a traffic management system at an intersection. This enables the local authorities to apply the communication systems of traffic and infrastructure data exchange as traffic light detectors for special times such as times of building works and detours. The modular approach also enables a good possibility towards future requirements. The device technology of traffic light control systems has a long application duration and life span which also leads towards a modular device which is easily updatable and programmable. The traffic light control systems in use may not

have been designed to calculate superior traffic flows out of wireless delivered FCD. Therefore the modular system has to be able to process additional data.

2.2 Requirements Conclusion

Putting together the requirements of a co-operative traffic light system into a solution, the decision was made to use two different communication systems in order to set up a co-operative traffic light system enabling traffic data exchange with participating vehicles of the traffic. On the side of the traffic, an application for transferring vehicle data such as speed and position was designed for a customary PDA. Such PDAs equipped with GPS sensors are already widely-used for navigation applications. They also offer operating systems which make it possible to develop a WLAN based communication interface. This vehicle-sided application was developed in Java and offers vehicle announcement at the communication unit of the traffic lights and furthermore self locating algorithms in order to deliver positioning and speed data to the communication unit.

The modular communication unit for the communication between the traffic light to the traffic is a new set-up which is suitable for connecting the traffic light signal and for updating standardised interfaces. The communication interface is realised by the concept of a Set Top Box (STB) for traffic light control systems.

3 The Set Top Box: A Central Communication Unit for Co-operative Traffic Light Control Systems

The STB works as a communication mediator and data analysing unit of an intersection managing traffic light control systems between the traffic light system and the traffic flows of approaching vehicles.

The vehicles announce themselves at the STB and transmit information about their own speed, position and the intended direction through a customary PDA.
One STB at each intersection is located in the control cabinet of the traffic light system. Using the received individual vehicle data such as position and speed, it calculates the traffic flow data concerning the traffic flow density and flows for each lane of the intersection. The STB is connected to the traffic light device via a standardised LAN connection. As a virtual detector it informs the traffic light control systems about approaching traffic flows and optimisation of switching programs. The traffic light can therefore react better to topical flows of traffic and this allows a more dynamic control of the road traffic.

Furthermore the STB can establish a connection to the traffic calculator of the traffic management centrals. The local traffic information can then be used to create global traffic management strategies due to more distinctive information concerning the traffic flows at crossings. Here the STB enables the feedback from local detectors towards an overall traffic management.

Several Set Top Boxes can furthermore be interconnected to each other by WLAN or UMTS communications centralised using a server or even decentralised. Via this

interconnection, information about traffic flow movement and changes can be communicated and used for adaptive traffic light switching such as adaptive progressive signal systems.

3.1 Hardware Setup of the STB

The main hardware of the Set Top Box is based on a specialised industrial PC with a custom-made board. So far, the communication to the vehicles is established using a low-cost personal digital assistant (PDA) which enables WLAN-communication (802.11) by an access point. The communication is suitable for vehicle speeds about 50km/h as they are due in areas of urban intersections. These PDAs are mostly equipped with GPS-devices already what makes them usable for a use as navigation systems in cars. The communication from the STB to PDAs enables a vehicle type independent communication.

Fig. 1. Set Top Box.

One of the main advantages of such a box as it is shown in figure 1 is its flexibility because it can be built in wherever and for as long as needed. It just requires a traffic light system. The system of the traffic light control stays untouched, therefore the usage of a STB does not threat the safety of an already running traffic light system. Therefore certifications of the traffic light control program are not altered either. The box does not interfere with the set control programs, but can be used to optimise their switching. The STB can be adapted to the interfaces of the manufacturer-specific control devices. If many traffic lights are equipped with these boxes as it is shown in

figure 2, the traffic information can be gathered and used for a broad area for the needed time. Because of these manifold usages, the financial costs for the local authorities can be reduced by additionally enabling a forward-looking traffic detection and traffic management.

Fig. 2. Set Top Box as it is placed in a control cabinet of traffic lights.

3.2 Realised Functionalities

Regarding the requirements mentioned above several functionalities for the communication between vehicles (PDA) and STB and between traffic lights to STB are realised. An outline of the communication between Set Top Box, traffic lights and vehicles' PDA is shown in figure 3.

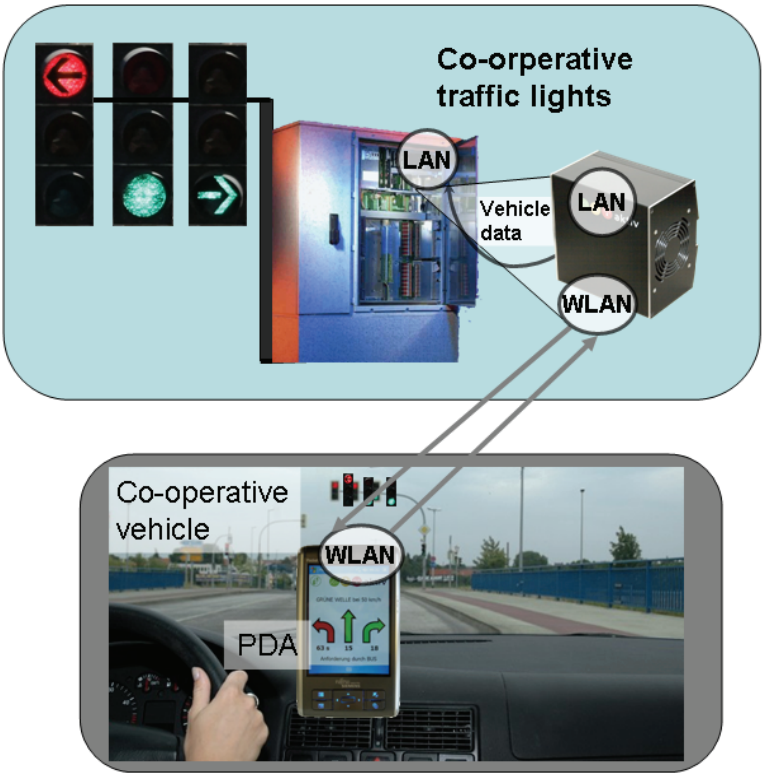


Fig. 3. Communication established by STB.

For the drivers following information based on traffic flow and traffic light system data, the following is delivered from the STB to the PDA in the vehicles:

- Transfer of cycle times of the red light and estimated waiting times till the vehicle's crossing of the intersection
- Speed recommendations in order to achieve the best travel time e.g. due to progressive signal systems
- General routing information about speed limits or congestions
- Information in cases of rapid traffic light control systems programming change e.g. because of passing emergency services

In return, the PDAs transfer the following data to the STB to be calculated and used for adaptive traffic light switching:

- Current speed and positions, especially to identify the single (heading) lanes
- If necessary the declaration of turning at the intersection

The STB connected to the control unit of the traffic light system computes the traffic flows at the intersection and offers the following detector data to the unit:

- Offering recommendations concerning switching status

Because of possible interconnections between several STBs at different crossroads and the traffic management centrals, the exchange of topical data can be used to develop and realise global traffic management strategies.

4 Conclusions and Future Steps

The approach of a modular car-to-infrastructure communication presented here enables the combination of the advantages of microscopic and macroscopic traffic management: local traffic adaptation and global ease of traffic flows.

The prototypical solution could be proposed by the development of a Set Top Box (STB), a specialised industrial PC with LAN and WLAN communication. The box is the central communication unit, connected to a control cabinet of a traffic light system at one side and via WLAN to standard PDAs in vehicles on the other side. The STB detects traffic flows and optimal switching strategies which can be put into reality by the connected traffic light and management systems.

This vehicle – STB – traffic light communication establishes a dynamic and co-operative control of the traffic flows. The communication could be established and the main traffic and control data was transferred.

The currently used proprietary communication protocols are being tested and validated in various field tests. Tests concerning the ability of the WLAN connection during various vehicle speeds, different environmental conditions such as building development or radio noise are currently being carried out. Furthermore issues of data security have to be addressed as well as the establishment of the currently developed standard for the car-to-car communication WLAN 802.11p.

The future steps will address the interaction of navigation systems in the concept of co-operative traffic lights. Then on the one hand the navigation system users could also be aware of the current states of the traffic light and the routing could take into account possible hold-ups caused by traffic lights. On the other hand traffic flows could be predicted in order to optimise the cycle times of traffic lights. This enables smooth traffic flows in spite of more participants of the individual traffic.

Furthermore the Set Top Boxes can be used as Road Side Units 10 also enabling car to car communication 11. The applications inside the car could be updated in order to acquire more driving concerning data such as road conditions or traffic situations. The data would be transferred to and computed by the STBs. The traffic information can then be used by the traffic management centrals or decentralised by distributing it to the passing vehicles. The usage of low-cost standard vehicle equipment could achieve a high coverage of communicating cars easily which is a central key point in encouraging co-operative car-to-car and of course car-to-infrastructure technologies.

Acknowledgements

The research and development work in the project AKTIV is funded by the Ministry of Education and Research (BMBF), grant number 19 P 6018 N.

References

1. Czogalla, O.; Hoyer, R.: Simulation based Design of Control Strategies for urban Traffic Management and Control. 4th World Congress on Intelligent Transport Systems 97, Berlin, October 21 24, 1997
2. Czogalla, O.; Hoyer, R.: Model based approximation of traffic actuated signal control for mesoscopic traffic simulation. 6th World Congress on Intelligent Transport Systems, Toronto, Nov 8-11, 1999
3. Lämmer S.: Reglerentwurf zur dezentralen Online-Steuerung von Lichtsignalanlagen in Straßennetzwerken“, PHD-thesis (Dissertation), Technische Universität Dresden, (2007)
4. Hoyer, R.; Herrmann, A.; Schönrock, R.: Vehicle actuated traffic lights – a challenge to modelling of inner-city networks, In: Proceedings of the 13th World Congress on Intelligent Transport Systems, London, 8-12 October, 2006
5. Richtlinien für Lichtsignalanlagen an Straßen (RiLSA), Forschungsgesellschaft für Straßen- und Verkehrswesen, Konrad-Adenauer-Straße 13, 50996 Köln, 1992
6. Mück, J.: Schätzverfahren für den Verkehrszustand an Lichtsignalanlagen unter Verwendung haltliniennaher Detektoren. Tagungsband „Heureka 2002“, Karlsruhe, 2002
7. Priemer, C.; Friedrich, B.: Optimierung von modellierten Warteprozessen im Rahmen adaptiver Netzsteuerungen durch C2I – Daten, Tagungsband „Heureka 2008“, Hrsg. Forschungsgesellschaft für Straßen- und Verkehrswesen, Köln, 2008
8. Boltze M., Reusswig A.: Bessere Lichtsignalanlagen mit System PA, NORAMA, thema FORSCHUNG (2/2006), 88
9. Pohlmann T., Naumann S.: Standardized Device and Interfaces for Traffic Light Controllers, In Proceedings of the 15th World Congress on Intelligent Transport Systems, New York, 16-20 November (2008)
10. Lochert C., Scheuermann B., Wewetzer C., Luebke A., Mauve M.: Data Aggregation and Roadside Unit Placement for a VANET Traffic Information System, VANET 2008: Proceedings of the Fifth ACM International Workshop on VehiculAr Inter-NETworking, San Francisco, CA, USA, (2008), 58-65
11. Stephan Eichler, Christoph Schroth, Jörg Eberspächer Car-to-Car Communication In Proceedings of the VDE-Kongress - Innovations for Europe, 2006 Aachen, Germany

An Adaptive Cruise Control System based on Self-Learning Algorithm for Driver Characteristics

Lei Zhang, Jianqiang Wang and Keqiang Li

State Key Laboratory of Automotive Safety and Energy
Tsinghua University, Beijing 100084, China

Abstract. An Adaptive Cruise Control system prototype based on self-learning algorithm for driver characteristics is presented. To imitate the driver operations during car-following, a driver model is developed to generate the desired throttle depression and braking pressure. A self-learning algorithm for driver characteristics is proposed based on the Recursive Least Square method with forgetting factor. Using this algorithm, the parameters of the driver model are real-time identified from the data sequences collected during the driver manual operation state, and the identification result is applied during the system automatic control state. The system is verified in a driving assistance system testbed with electronic throttle and electro-hydraulic brake actuators. The experimental results show that the self-learning algorithm is effective and the system performance is adaptive to driver characteristics.

1 Introduction

With the traffic density increasing rapidly, car-following has become the most frequent driving scenario to the driver. In the vehicle active safety field, several types of driving assistance systems have been actualized for the car-following scenario such as Adaptive Cruise Control (ACC) [1], Stop & Go (S&G) [2] and Forward Collision Warning/Avoidance (FCW/FCA) [3]. The aims of the systems are to facilitate driver to maintain a safe and comfortable car-following state or to mitigate the workload of the driver [4]. Because of the interaction between the driver and the assistance system, the driver behavior and characteristics during car-following have been considered as important issues in system development.

The research on modeling driver behavior in car-following scenario dates back to the 1950s and many types of models were established with different approaches [5]. The classical method is using mathematic functions to represent the relationship between variables like host vehicle speed, acceleration, relative speed and distance headway, such as the Gazis-Herman-Rothery (GHR) model [6], the Gipps model [7] and the linear (Helly) model [8]. These models can be applied to the system control algorithm, but as the required outputs of the models are the desired vehicle motion states, complicated vehicle dynamics model needs to be added. Some models are designed to imitate the driver's throttle and braking operations directly [9]. This method could avoid the vehicle dynamics problem such as the inverse model of vehicle longitudinal

dynamics. However, the parameters of these models are fixed during system operation and cannot be adaptive to individual driver car-following characteristics.
In this paper, a driver model is proposed to imitate throttle and braking operations of the driver and a self-learning algorithm for driver characteristics is designed based on Recursive Least Square (RLS) method with forgetting factor. Using this algorithm, the parameters of the driver model can be real-time identified from the data sequences collected during manual driving operation state, and the identification result is applied during the system automatic control state. The driver model and the self-learning algorithm are implemented in a driving assistance system test-bed and the functions of the system are validated by tests in real traffic.

2 Driver Behavior Test and Characteristics Analysis

The driver behavior during car-following is a significant factor for the development of driving assistance system. To investigate essential driver characteristics and establish driver behavior database, driver behavior tests in real traffic environment are executed and the signals including host vehicle speed, acceleration, depression of accelerator pedal/throttle, braking pressure, relative distance/speed to leading vehicle, and GPS information are recorded with 10Hz data capture frequency. Thirty drivers are invited as experimental subjects to drive on the city highway for 1 hour per person. The drivers are suggested to drive freely according to their own styles and habits. The data sequences of steady car-following behavior, which corresponds to the ACC function, are extracted from the test data. This behavior is defined as that the driver controls the host vehicle to follow a constant leading vehicle steadily more than 15 seconds without braking and lane-changing. Two common variables are discussed in the data analysis to describe driver characteristics. One is Time Headway (*THW*):

$$THW = \frac{D}{v} \tag{1}$$

The other one is Time-to-Collision (*TTC*, and its inverse *TTCi*):

$$TTC = \frac{D}{v_r}, TTCi = \frac{v_r}{D} \tag{2}$$

Where: D is the distance between the host vehicle and the leading vehicle; v is the host speed vehicle; and v_r is the host vehicle's relative speed to the leading vehicle. The frequency contour of *THW* and *TTCi* of one driver's steady car-following behavior is shown in Fig 1. The number on each area border (50%, 75%, 95% and 99%) in this figure means the percentage of the data points falling inside this border. It is clear that 50% of *THW* and *TTCi* data distribute in a relatively concentrated area where *THW* is around 1.2s to 2.6s and *TTCi* is around -0.05 to 0.05s^{-1}. This phenomenon indicates that the driver prefers to keep *THW* and *TTCi* in specific ranges, and these two variables can be considered as the driver control targets during car-following for the driver model design.

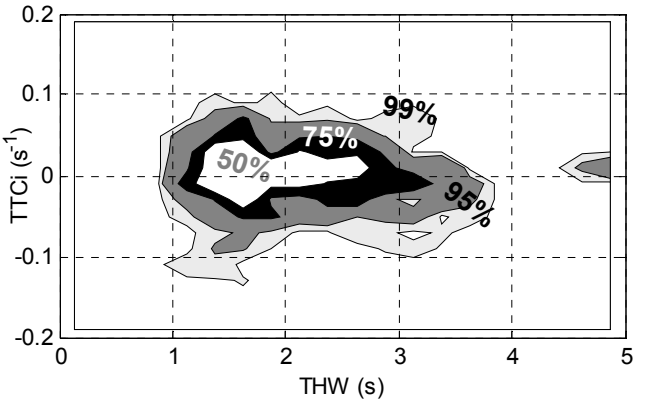


Fig. 1. Frequency contour of *THW* and *TTCi*, steady car-following behavior, one driver.

3 The System Control Strategy

3.1 Control Strategy Structure

The structure of the ACC system control strategy is shown in Fig 2. The upper controller is a driver model to imitate the driver's operation. The inputs of the model are the motion states of the leading vehicle and the host vehicle, and the outputs include desired throttle depression Th_{des} and desired braking pressure Pb_{des}. The lower controller makes the electronic throttle and Electro-Hydraulic Brake (EHB) actuators follow the desired control variables with PID (Proportion-Integral-Differential) control algorithm.

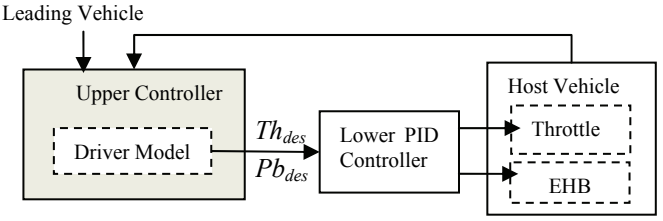


Fig. 2. The structure of the ACC system control strategy.

3.2 Driver Model

According to the driver characteristics analysis of steady car-following behavior, the driver desires to obtain *THW* in his/her preferred range and *TTCi* around zero. This control target is implemented based on the variations of throttle depression and brake pressure. If *THW* and *TTCi* reach desired values and the leading vehicle drives in a constant speed, the driver will fix the throttle and keep the current speed. Based on

this analysis, a driver model is proposed:

$$p_{des}(t) = Th_{ss}(t) + K_{THW} \cdot [THW(t) - THW_d] + C_{TTCi} \cdot TTCi(t) \tag{3}$$

Where: $P_{des}(t)$ is generalized depression at time t; $Th_{ss}(t)$ is steady throttle depression to keep the current host vehicle speed $v(t)$; THW_d is the driver's desired time headway; K_{THW} and C_{TTCi} are error gains of THW and $TTCi$ respectively.
Interpolation method is used for Th_{ss} calculation based on the experimental calibration. The desired control variables, Th_{des} and Pb_{des}, are calculated according to the value of the generalized depression p_{des}. The throttle depression for idle-speed is 15.
When $p_{des}(t)$ >15:

$$\begin{cases} Th_{des}(t) = p_{des}(t) \\ Pb_{des}(t) = 0 \end{cases} \tag{4}$$

Considering the driver's operation delay at the switching between accelerator and brake pedal, the braking control is not activated immediately when $p_{des}(t)$ falls below the idle-speed depression 15. When 15>= $p_{des}(t)$ >10:

$$\begin{cases} Th_{des}(t) = 15 \\ Pb_{des}(t) = 0 \end{cases} \tag{5}$$

When $p_{des}(t)$ <=10:

$$\begin{cases} Th_{des}(t) = 15 \\ Pb_{des}(t) = B_{pb} \cdot [p_{des}(t) - 10] \end{cases} \tag{6}$$

Where: B_{pb} is the gain from p_{des} to Pb_{des}, whose value is set as -0.1, and the unit of the desired brake pressure Pb_{des} is MPa. The maximal value of Pb_{des} is set as 10MPa.

4 Self -Learning Algorithm for Driver Characteristics

The driver model could describe the driver characteristics and present the individual differences during car-following. The parameter THW_d presents the driver's preferred following distance at same vehicle speed level and reflect his/her aggressive degree. The parameters K_{THW} and C_{TTCi} present the driver's sensitivity of THW error and $TTCi$ error. To improve the system's adaptability of individual driver characteristics, a self-learning algorithm based on Recursive Least Square (RLS) method is proposed. The core idea of this algorithm is to identify the model parameters from the driver manual car-following drive state on-line and apply the identification result to the model during system automatic driver state. Because of the time-variability of the driver, it is supposed that the latest data of driver operation will describe the driver characteristics more accurately and therefore, forgetting factor is brought into the algorithm. The flow chart of this self-learning algorithm is shown in Fig 3.

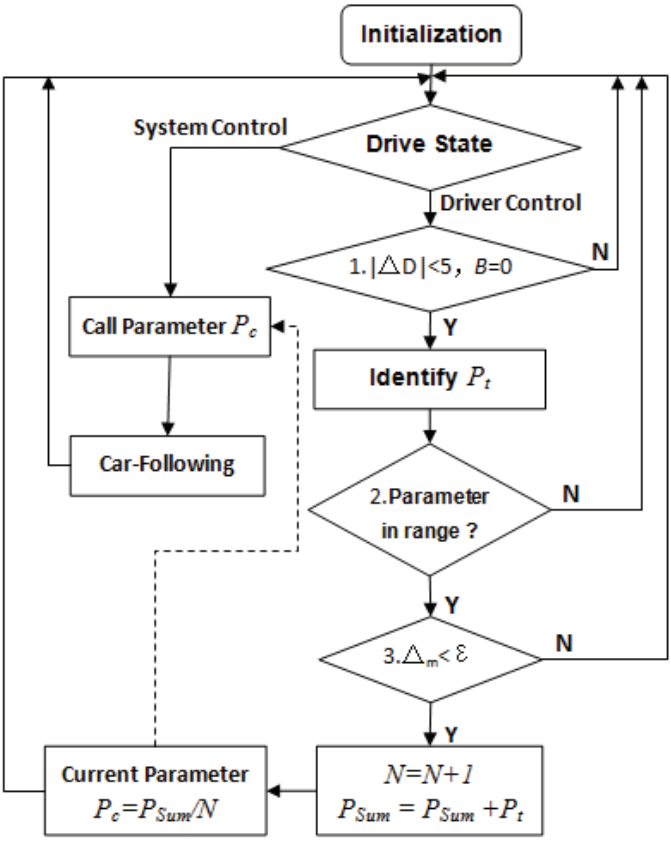


Fig. 3. The flow chart of self-learning algorithm.

After the system initialization, the signal collection of distance D, relative speed v_r, host vehicle speed v and throttle depression Th is enabled. The driver selects the drive states. During the driver manual control process, the algorithm starts the cycle to judge the car-following state and identify the parameters step-by-step. The system step length is 0.1s. The parameters THW_d, K_{THW} and C_{TTCi} are identified from steady car-following data sequence.

The first condition is that the leading vehicle should be a constant target (i.e. no target changing such as cut-in and cut-out scenarios) and this condition is judged according to the variation of the distance signal. Furthermore, the driver is not controlling the brake system. At step k:

$$\begin{cases} \Delta D = \left| D(k) - D(k-1) \right| < 5 \\ B(k) = 0 \end{cases} \tag{7}$$

If the first condition is satisfied, the algorithm will use the current data $D(k)$, $v_r(k)$, $v(k)$ and $Th(k)$ to start the iteration process of LRS method.

The observation vector of the iteration process is $\boldsymbol{h}^T(k)$:

$$\boldsymbol{h}^T(k)=[\frac{D(k)}{v(k)} \quad -1 \quad \frac{v_r(k)}{D(k)}] \tag{8}$$

The output of the process is $z(k)$:

$$z(k)=Th(k)-Th_{ss}(k) \tag{9}$$

Where: $Th_{ss}(k)$ is the current steady throttle which can be interpolated with $v(k)$. According to the standard linear square form, the parameter vector $\hat{\boldsymbol{\theta}}(k)$ to identify in this process can be derived from Equation (**1**):

$$\hat{\boldsymbol{\theta}}(k)=[\hat{\theta}_1(k)\ \hat{\theta}_2(k)\ \hat{\theta}_3(k)]^T=[K_{THW}(k)\ \ K_{THW}(k)\cdot THW_d(k)\ \ C_{TTCi}(k)]^T \tag{10}$$

The iteration algorithm of LRS method with forgetting factor is [10]:

$$\begin{aligned}\hat{\boldsymbol{\theta}}(k)&=[\hat{\theta}_1(k)\ \hat{\theta}_2(k)\ \hat{\theta}_3(k)]^T=[K_{THW}(k)\ \ K_{THW}(k)\cdot THW_d(k)\ \ C_{TTCi}(k)]^T\\ \boldsymbol{K}(k)&=\boldsymbol{Q}(k-1)\boldsymbol{h}(k)[\boldsymbol{h}^T(k)\boldsymbol{Q}(k-1)\boldsymbol{h}(k)+\mu]^{-1}\\ \boldsymbol{Q}(k)&=\frac{1}{\mu}[\boldsymbol{I}-\boldsymbol{K}(k)\boldsymbol{h}^T(k)]\boldsymbol{Q}(k-1)\end{aligned} \tag{11}$$

Where: $\boldsymbol{K}(k)$ and $\boldsymbol{Q}(k)$ are process matrices and μ is forgetting factor with value 0.9. The identified parameter vector of the driver model in this step is $P_t(k)$:

$$\begin{aligned}\boldsymbol{P}_t(k)&=[THW_d(k)\ \ K_{THW}(k)\ \ C_{TTCi}(k)]^T\\ &=[\hat{\theta}_2(k)/\hat{\theta}_1(k)\ \ \hat{\theta}_1(k)\ \ \hat{\theta}_3(k)]^T\end{aligned} \tag{12}$$

After obtaining $\boldsymbol{P}_t(k)$, the second condition is that the parameters should be in proper ranges. These ranges are provided by the off-line parameter identification results of the driver real traffic steady car-following data sequences with linear square method, which are shown in Table 1. The proper range of each parameter is selected as its 25% to 75% accumulation frequency.

Table 1. Data Statistics of Driver Model Parameters.

	Mean	Std	Max	Min	25%	75%
THW_d	1.80	3.18	56.51	0.15	0.9	2.3
K_{THW}	44.26	50.68	408.35	0.10	6	95
C_{TTCi}	-157.3	129.4	-0.96	-842.7	-20	-300

When the identified $\boldsymbol{P}_t(k)$ is in the proper ranges, the parameters will be inspected by the third condition to judge if the identified result tends to be steady correspondingly:

$$\Delta_m=\max(\Delta_{THW}(k),\Delta_K(k),\Delta_C(k))<\varepsilon \tag{13}$$

Where:

$$\Delta_{THW}(k) = \left| \frac{THW_d(k) - THW_d(k-1)}{THW_d(k)} \right| \tag{14}$$

$$\Delta_K(k) = \left| \frac{K_{THW}(k) - K_{THW}(k-1)}{K_{THW}(k)} \right| \tag{15}$$

$$\Delta_C(k) = \left| \frac{C_{TTCi}(k) - C_{TTCi}(k-1)}{C_{TTCi}(k)} \right| \tag{16}$$

ε is the threshold, which is 0.5% in this algorithm.

Because that the driver state is time-varied, the identified parameters are always fluctuating. In order to find the parameters describing the driver characteristics as precisely as possible, an accumulation method is used:

$$\boldsymbol{P}_{sum} = \boldsymbol{P}_{sum} + \boldsymbol{P}_t(k) \tag{17}$$

All parameters satisfied the conditions are accumulated to $\boldsymbol{P}_{sum}$ and when the drive state switches to system automatic driving, the current parameter vector $\boldsymbol{P}_c$ is called by the driver model:

$$\boldsymbol{P}_c = \frac{\boldsymbol{P}_{sum}}{N} \tag{18}$$

Where: N is the counter of the parameters.

With the running time increasing, the algorithm will accumulate more identified results from driver manual operation and the learning effect will be improved. The driver model will be closer to the driver average characteristics. During the algorithm running process, if any of the three conditions are not satisfied, the iteration will be stopped and the current $\boldsymbol{P}_{sum}$ and N will be held. Until new proper parameters are identified, the accumulation will be continued.

5 System Verification in Driving Assistance System Test-bed

A test-bed on a passenger car is developed to verify the system functions including driver characteristics self-learning algorithm and ACC. During the self-learning algorithm verification experiment, a driver subject drives the test-bed vehicle in real traffic and the self-learning algorithm runs online synchronously to identify the model parameters. The parameter identification test continues for 600 seconds to make the results closer to the driver average characteristics, Fig 4 gives the driver manual operation data sequence when following a specified leading vehicle. Fig 5 shows the parameter identification process from this data sequence. It is indicated that the algorithm is effective and the parameters tend to be stable gradually after some fluctuation at the beginning. At the end of the test, the final identification results are: THW_d = 1.84, K_{THW} = 33.5, C_{TTCi} = -109.5.

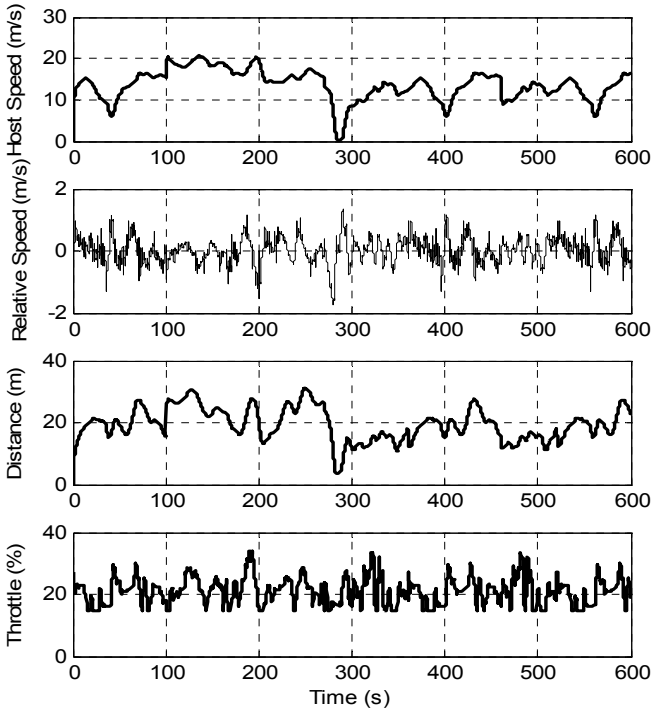


Fig. 4. Data sequences of driver manual car-following.

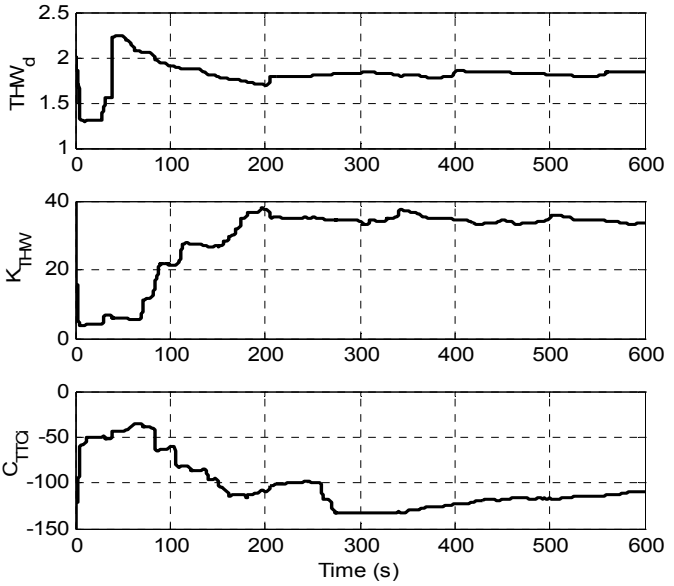


Fig. 5. Driver model parameters on-line identification result.

Using these identified parameters, the system is switched to ACC mode and Fig 6 shows a data sequence of system automatic car-following. The system can track the leading vehicle's speed steadily and keep safety distance. The control performances of the upper and lower controllers are both favorable.

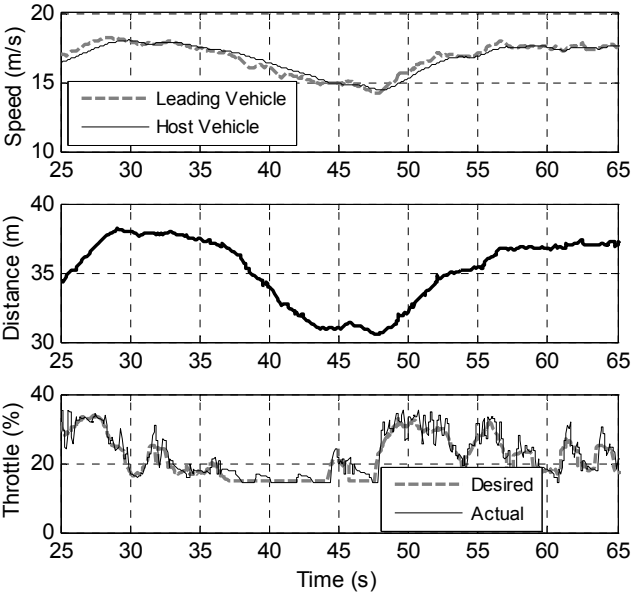


Fig. 6. The performance of the system ACC function.

More experiments of ACC verification are carried out in real traffic and the system performance is analyzed with *THW*-*TTCi* frequency contour, which is shown in Fig 7.

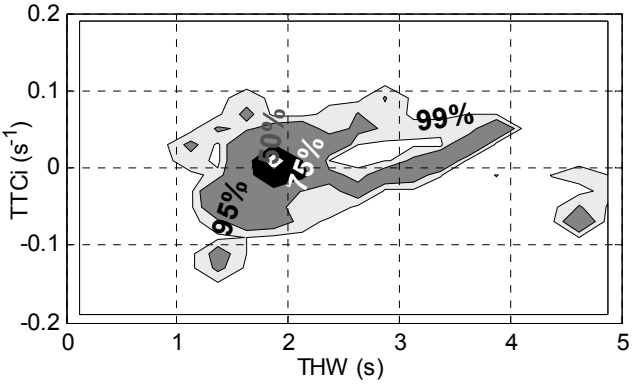


Fig. 7. Frequency contour of *THW* and *TTCi* during system control.

Comparing with Fig.1, it is indicated that the overall data distributions (99% percentage) of the system and the driver are similar. Based on the parameter identified from the driver behavior, the system performance is adaptive to the driver characteristics

and gives the driver comfortable riding experience. Furthermore, the 50% and 75% areas of system performance are more centralized than the driver. This result indicates that the *THW* and *TTCi* fluctuations during system control state are much smaller and the system is more stable than the driver.

6 Conclusions

In this paper, an Adaptive Cruise Control system prototype with self-learning functions is developed on a passenger car test-bed.
(1) Driver real traffic tests are carried out and the driver behavior database for the system upper controller design is established. The data analysis of steady car-following show that the driver prefers to keep *THW* and *TTCi* in specific ranges, and a driver model is designed based on this result.
(2) The Recursive Least Square method with forgetting factor can identify the driver model parameters online from data sequence of driver manual operation state, and the self-learning algorithm for driver characteristics is proposed with this method.
(3) The experimental results show that the ACC system can be adaptive to the driver characteristics automatically with the learned parameters. The system has similar performance with the driver manual operation and favorable acceptability of driver.

References

1. Ohno, H.: Analysis and modeling of human driving behaviors using adaptive cruise control. Industrial Electronics Society, IECON 2000, 26th Annual Conference of the IEEE, Vol.4, 2803-2808 (2000)
2. Yi, K., Hong, J., Kwon, Y. D.: A Vehicle Control Algorithm for Stop-and-Go Cruise Control. Proceedings of the Institution of Mechanical Engineers. Part D, Journal of automobile engineering, Vol. 215, 1099-1115 (2001)
3. Lemaire, E., El Koursi, E.M., Deloof, P., Ghys, J. P.: Safety Analysis of a Frontal Collision Warning System. IEEE Intelligent Vehicles Symposium, Vol. 2, 453 – 458 (2002)
4. Yoshida, T., Kuroda, H., Nishigaito, T.: Adaptive Driver-assistance Systems. HITACHI Review (2004)
5. Mark Brackstone, Mike McDonald: Car-Following: A Historical Review. Transportation Research Part F, Vol. 2: 181~196 (1999)
6. Gazis, D. C., Herman, R., & Rothery, R. W.: Nonlinear follow the leader models of traffic flow, Operations Research, Vol. 9, 545-567 (1961)
7. Gipps, P. G.: A Behavioral Car Following Model for Computer Simulation. Transportation Research B, Vol.15: 105-111 (1981)
8. Helly, W.: Simulation of Bottlenecks in Single Lane Traffic Flow. In Proceedings of the Symposium on Theory of Traffic Flow, Research Laboratories, General Motors, 207-238 (1961)
9. Erwin R. B., Nicholas J. W., Michael, P. M., Nobuyuki K.: Driver-Model-Based Assessment of Behavioral Adaptation. Proceedings. JSAE Annual Congress (2005)
10. Enso Ikonen, Kaddour Najim Advanced Process Identification and Control, CRC Press, 2002, ISBN 082470648X

Order-based Freight Transportation Operation Planning in Railway Networks

Davide Anghinolfi, Massimo Paolucci, Simona Sacone and Silvia Siri

Dept. of Communication, Computer and System Sciences
Via Opera Pia 13, 16145 Genova, University of Genova, Italy
{davide.anghinolfi,massimo.paolucci,simona.sacone,silvia.siri}
@unige.it

Abstract. In this paper we propose a planning procedure for serving freight transportation requests (i.e. orders) in a railway network in which the terminals are provided with innovative transfer systems. We consider a consolidated transportation system where different customers make their own requests for moving boxes (either containers or swap bodies) between different origins and destinations, with specific requirements on delivery times. The decisions to be taken concern the path (and the corresponding sequence of trains) that each box follows over the network and the assignment of boxes to train wagons, taking into account that boxes can change more than one train during their path and that train timetables are fixed. This planning problem is divided in two sequential phases: a preprocessing analysis for which a specific algorithm is provided and an optimization phase for which a mathematical programming formulation is proposed. The effectiveness of the proposed procedure is tested on a set of randomly generated instances.

1 Introduction

This work deals with the definition of a planning procedure for a centralized decision maker that must provide a transportation service to different customers by using an available railway infrastructure. The transportation demand from customers is given by a set of orders characterized by a certain number of boxes (in general of different types), an origin, a destination and time delivery specifications. The transportation offer is a railway network in which the number of trains, their schedules and paths are fixed. This railway network is innovative since the terminals are supposed to be equipped with rapid (horizontal) transfer systems for containers and swap bodies. This implies that a container can move from an origin to a destination terminal by changing different trains on its path, as it happens to passengers. Moreover, a peculiar characteristic of the proposed approach is that boxes of the same order can follow different paths on the network.

In the literature it is possible to find many planning approaches for intermodal transportation systems involving modelling and optimization techniques, that can be classified according to the considered planning level, i.e. strategic, tactical and operational, as it is deeply described in [1] and [2]. The problem faced in this paper can be considered as an operational problem combining two important decision aspects. The former

decision concerns which path a box must follow on the network and in which terminals it must change the train. This aspect is also treated in tactical problems, i.e. service network design problems, as described in [3], in which the optimal paths are searched for aggregated cargo flows instead of single load units as in the present paper. The latter decision aspect deals with the assignment of boxes to wagons of the trains selected to transport them. This aspect has been treated in [4], where rapid transshipment yards are considered as in the present paper; moreover, in [5] a load planning problem, i.e. the assignment of containers to train slots, is treated. In addition, differently from [6] and [7], we assume the train scheduling and routing as given and fixed. Therefore, the problem described in this paper provides a new approach for a planning procedure in a railway network with rapid rail-rail transfers, in which, for each box, the decisions to be taken concern the route to cover, the trains and the wagons to be placed in. Even though all these aspects have been already treated in various works that can be found in the literature, the main novelty of this paper (a preliminary version of this work can be found in [8]) stands in the consideration of all these decision aspects together.

The planning procedure is divided in two sequential phases, so that the output data of the former phase are input data of the latter. In the former phase, called *preprocessing*, each order is considered separately: by taking into account the network structure, the timetables and routes of trains, the origin, destination and time requirements of orders, all the sequences of trains that can be used for serving the boxes of the considered order are computed. In the latter phase, called *optimal assignment*, the assignment of each box to a train sequence and to a wagon of the trains composing the sequence is obtained by considering all the sequences of trains for each order and by taking into account some specific data about boxes and wagons. This is the result of a specific mathematical programming problem.

These two phases of the proposed planning procedure are described, respectively, in Section 2 and in Section 3. The effectiveness of this procedure is then verified with some experimental tests reported in Section 4. The conclusions and future developments of the work are then described in Section 5.

2 Preprocessing

The railway network is described by means of a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ where nodes represent railway terminals and links are railway connections between terminals. The input data of the preprocessing are referred to nodes of the network, a set $\mathcal{R}$ of trains and a set $\mathcal{O}$ of orders:

δ_n fixed cost for handling one box at terminal $n \in \mathcal{N}$
ρ_n hourly cost for the storage of one box at terminal $n \in \mathcal{N}$
Nl_r number of links covered by train $r \in \mathcal{R}$
$\mathbf{L}_r$ vector $1 \times Nl_r$ indicating the path (as a sequence of links) covered by train $r \in \mathcal{R}$
$t^{dep}_{r,l}$ expected departure time on link l *in* $\mathbf{L}_r$ for train $r \in \mathcal{R}$
$t^{arr}_{r,l}$ expected arrival time on link l *in* $\mathbf{L}_r$ for train $r \in \mathcal{R}$
$n^{\mathbf{O}}_o$ origin railway terminal for order $o \in \mathcal{O}$
$n^{\mathbf{D}}_o$ destination railway terminal for order $o \in \mathcal{O}$
t^{in}_o time instant in which goods of order $o \in \mathcal{O}$ are ready at the origin node

$t^*_{min,o}$ minimum delivery time instant for order $o \in \mathcal{O}$
$t^*_{max,o}$ maximum delivery time instant for order $o \in \mathcal{O}$
Np_o number of alternative paths connecting the origin and destination nodes of order $o \in \mathcal{O}$
$Nl_{o,p}$ number of links of path $p = 1, \ldots, Np_o$ of order $o \in \mathcal{O}$
$\mathbf{L}_{o,p}$ vector $1 \times Nl_{o,p}$ indicating the sequence of links of path $p = 1, \ldots, Np_o$ of order $o \in \mathcal{O}$
$\zeta_{o,p}$ cost of path $p = 1, \ldots, Np_o$ of order $o \in \mathcal{O}$ (this cost is related to the priority of path p in comparison with the other paths of the same order; we set $\zeta_{o,p} = 1$ for the primary path, $\zeta_{o,p} = 1.1$ for the secondary path, and so on)

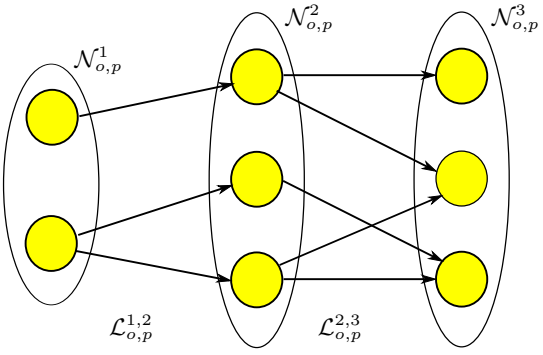


Fig. 1. Graph $\mathcal{G}_{o,p}$ in case $Nl_{o,p} = 3$.

The preprocessing phase identifies all the feasible train sequences for each order by analysing one order at a time and, for each order, each path singularly. Then, proceeding backward from the last link of a path, i.e. the last element of vector $\mathbf{L}_{o,p}$, to the first link, it is necessary to verify whether a train can be used on the considered link. For the last link of a path this is obtained finding all trains arriving in time with respect to the maximum delivery time $t^*_{max,o}$ and leaving not before the time t^{in}_o in which goods are ready at the origin. Going backward, considering the trains selected for a link, we search for those trains in the previous link such that the time connections are respected and, again, the departure time is not before t^{in}_o.

For each order and for each of the Np_o paths of the order, this procedure leads to the creation of a graph composed of $Nl_{o,p}$ partitions. Let us define this graph as $\mathcal{G}_{o,p} = (\mathcal{N}_{o,p}, \mathcal{L}_{o,p})$. As shown for example in Fig. 1 for $Nl_{o,p} = 3$, we denote the set of nodes in the different partitions as $\mathcal{N}^i_{o,p}, i = 1, \ldots Nl_{o,p}$, and the set of links as $\mathcal{L}^{i,i+1}_{o,p}$, $i = 1, \ldots, Nl_{o,p} - 1$. Each node in $\mathcal{N}^i_{o,p}$ is a feasible train in the i-th railway link of path p of order o and each link of $\mathcal{L}^{i,i+1}_{o,p}$ represents the fact that the connected nodes (i.e. trains) are adjacent in a train sequence. For this reason, after creating this graph, all the train sequences are obtained as all the possible paths in this graph. In the following, the backward procedure for the construction of graph $\mathcal{G}_{o,p}$ for order o and for a given path $p \in \{1, \ldots, Np_o\}$ is described. In this procedure the constant Δt

represents the time necessary for a box unloaded from a train in a terminal to be ready for being loaded on another train, and $\mathcal{R}_l$ represents the set of trains covering link l.

initialize $m = Nl_{o,p}$;
set l as the last link of the path;
foreach r *in* $\mathcal{R}_l$ **do**
  if $t^{arr}_{r,l} \leq t^{*}_{max,o}$ **and** $t^{dep}_{r,l} \geq t^{in}_{o}$ **then**
    add r in $\mathcal{N}^{m}_{o,p}$;
  end
end
foreach ρ *in* $\mathcal{N}^{m}_{o,p}$ **do**
  set $t^{*}_{\rho,m} = t^{dep}_{\rho,l} - \Delta t$
end
for *i=m-1* **to** *1* **do**
  set l as the i-th link of the path;
  foreach ρ *in* $\mathcal{N}^{i+1}_{o,p}$ **do**
    foreach r *in* $\mathcal{R}_l$ **do**
      if $t^{arr}_{r,l} \leq t^{*}_{\rho,m}$ **and** $t^{dep}_{r,l} \geq t^{in}_{o}$ **then**
        add r in $\mathcal{N}^{i}_{o,p}$;
        add link (r, ρ) in $\mathcal{L}^{i,i+1}_{o,p}$;
      end
    end
    foreach ρ *in* $\mathcal{N}^{i}_{o,p}$ **do**
      set $t^{*}_{\rho,i} = t^{dep}_{\rho,l} - \Delta t$
    end
  end
end

After the completion of the backward procedure constructing the graph $\mathcal{G}_{o,p}$, a forward procedure is applied which, for each order o and for each path p, $p = 1, \ldots, Np_o$, identifies $Ns_{o,p}$ train sequences. Each train sequence $\mathbf{S}_{o,p,s}, s = 1, \ldots, Ns_{o,p}$, is a vector $1 \times Nl_{o,p}$ listing the trains for each link of path p for order o. While proceeding forward in the graph, the cost associated with each train sequence is computed. For each train sequence, let $\varphi^{n}_{o,p,s} \in \{0, 1\}$ denote whether the box changes train in terminal $n \in \mathcal{N}$ (without considering the origin and destination terminal) and $H^{n}_{o,p,s}$ denote the time (in hours) in which a box is stored at terminal $n \in \mathcal{N}$. Then the cost $C_{o,p,s}$ for the sequence s of path p for order o is computed as follows:

$$C_{o,p,s} = \zeta_{o,p} \sum_{i=1}^{n} \left(\varphi^{n}_{o,p,s} \cdot \delta_n + H^{n}_{o,p,s} \cdot \rho_n \right) \tag{1}$$

Finally, we need to define the following sets to state the planning problem described in Section 3:

– $\mathcal{R}_{o,p,s},\ p = 1, \ldots, Np_o,\ s = 1, \ldots, Ns_{o,p}$, is the set of trains included in sequence s of path p for order o;

- $\mathcal{S}^U_{o,p,n,r}$, $p = 1, \ldots, Np_o$, $n \in \mathcal{N}$, $r \in \mathcal{R}_n$ is the set gathering the indices of sequences of path p for order o so that boxes are unloaded from train r at node n (by definition, it is $\mathcal{S}^U_{o,p,n,r} \subseteq \{1, \ldots, Ns_{o,p}\}$);
- $\mathcal{S}^L_{o,p,n,r}$, $p = 1, \ldots, Np_o$, $n \in \mathcal{N}$, $r \in \mathcal{R}_n$ is analogous to the previous set but refers to loading operations;
- $\mathcal{S}^T_{o,p,n,r}$, $p = 1, \ldots, Np_o$, $n \in \mathcal{N}$, $r \in \mathcal{R}_n$ refers to transfer operations (i.e. trains that pass a given node and do not involve any loading or unloading operations).

3 Optimal Assignment

The definition of the optimization problem is based on the output data of the preprocessing phase as well as the following input data:

Nb_o number of boxes associated with order $o \in \mathcal{O}$
$\pi_{o,b}$ length of box $b = 1, \ldots, Nb_o$ of order $o \in \mathcal{O}$
$\omega_{o,b}$ weight of box $b = 1, \ldots, Nb_o$ of order $o \in \mathcal{O}$
Ω_r maximum bearable weight for train $r \in \mathcal{R}$
K_r cost related to the use of train $r \in \mathcal{R}$
Nw_r number of wagons of train $r \in \mathcal{R}$
$\Omega_{r,w}$ maximum bearable weight of wagon $w = 1, \ldots, Nw_r$ of train $r \in \mathcal{R}$
$\Pi_{r,w}$ length of wagon $w = 1, \ldots, Nw_r$ of train $r \in \mathcal{R}$
σ_n maximum number of handling operations (loading and unloading) for each train at terminal $n \in \mathcal{N}$ (this term depends on the handling capacity provided by each terminal)

The problem decision variables are listed in the following:

- $y_{o,b,p,s} \in \{0,1\}$, $o \in \mathcal{O}$, $b = 1, \ldots, Nb_o$, $p = 1, \ldots, Np_o$, $s = 1, \ldots, Ns_{o,p}$, assuming value 1 if box b of order o is assigned to sequence s of path p, otherwise equal to 0;
- $x_{o,b,p,s,r,w} \in \{0,1\}$, $o \in \mathcal{O}$, $b = 1, \ldots, Nb_o$, $p = 1, \ldots, Np_o$, $s = 1, \ldots, Ns_{o,p}$, $r \in \mathcal{R}_{o,p,s}$, $w = 1, \ldots, Nw_r$, assuming value 1 if box b of order o is assigned to wagon w of train r in sequence s of path p, otherwise equal to 0;
- $z_r \in \{0,1\}$, $r \in \mathcal{R}$, assuming value 1 if train r is used, otherwise equal to 0.

The planning problem is formulated as a 0/1 linear programming (LP) problem whose objective function considers the cost terms associated with train sequences (computed in the preprocessing phase) and train costs.

Problem 1.

$$\min_{\substack{y_{o,b,p,s} \\ x_{o,b,p,s,r,w} \\ z_r}} \sum_{o \in \mathcal{O}} \sum_{b=1}^{Nb_o} \sum_{p=1}^{Np_o} \sum_{s=1}^{Ns_{o,p}} C_{o,p,s} \cdot y_{o,b,p,s} + \sum_{r \in \mathcal{R}} K_r \cdot z_r \tag{2}$$

subject to

$$\sum_{p=1}^{Np_o} \sum_{s=1}^{Ns_{o,p}} y_{o,b,p,s} = 1 \qquad o \in \mathcal{O}\ b = 1, \ldots, Nb_o \tag{3}$$

$$\sum_{w=1}^{Nw_r} x_{o,b,p,s,r,w} = y_{o,b,p,s} \qquad o \in \mathcal{O}\ b = 1,\ldots,Nb_o\ p = 1,\ldots,Np_o$$
$$s = 1,\ldots,Ns_{o,p}\ r \in \mathcal{R}_{o,p,s} \quad (4)$$

$$\sum_{o\in\mathcal{O}}\sum_{b=1}^{Nb_o}\sum_{p=1}^{Np_o}\sum_{s\in(\mathcal{S}^L_{o,p,n,r}\cup\mathcal{S}^T_{o,p,n,r})}\sum_{w=1}^{Nw_r}\omega_{o,b}x_{o,b,p,s,r,w} \le \Omega_r z_r \qquad n \in \mathcal{N}\ r \in \mathcal{R}_n \quad (5)$$

$$\sum_{o\in\mathcal{O}}\sum_{b=1}^{Nb_o}\sum_{p=1}^{Np_o}\sum_{s\in(\mathcal{S}^L_{o,p,n,r}\cup\mathcal{S}^T_{o,p,n,r})}\pi_{o,b}x_{o,b,p,s,r,w} \le \Pi_{r,w}$$
$$n \in \mathcal{N}\ r \in \mathcal{R}_n\ w = 1,\ldots,Nw_r \quad (6)$$

$$\sum_{o\in\mathcal{O}}\sum_{b=1}^{Nb_o}\sum_{p=1}^{Np_o}\sum_{s\in(\mathcal{S}^L_{o,p,n,r}\cup\mathcal{S}^T_{o,p,n,r})}\omega_{o,b}x_{o,b,p,s,r,w} \le \Omega_{r,w}$$
$$n \in \mathcal{N}\ r \in \mathcal{R}_n\ w = 1,\ldots,Nw_r \quad (7)$$

$$\sum_{o\in\mathcal{O}}\sum_{b=1}^{Nb_o}\sum_{p=1}^{Np_o}\sum_{s\in(\mathcal{S}^L_{o,p,n,r}\cup\mathcal{S}^U_{o,p,n,r})}\sum_{w=1}^{Nw_r}x_{o,b,p,s,r,w} \le \sigma_n \qquad n \in \mathcal{N}\ r \in \mathcal{R}_n \quad (8)$$

$$y_{o,b,p,s} \in \{0,1\}$$
$$o \in \mathcal{O}\ b = 1,\ldots,Nb_o\ p = 1,\ldots,Np_o\ s = 1,\ldots,Ns_{o,p} \quad (9)$$

$$x_{o,b,p,s,r,w} \in \{0,1\} \qquad o \in \mathcal{O}\ b = 1,\ldots,Nb_o\ p = 1,\ldots,Np_o$$
$$s = 1,\ldots,Ns_{o,p}\ r \in \mathcal{R}_{o,p,s}\ w = 1,\ldots,Nw_r \quad (10)$$

$$z_r \in \{0,1\} \qquad r \in \mathcal{R} \quad (11)$$

Constraints (3) impose that each box of each order is assigned to one and only one train sequence, while (4) impose the relation between $y_{o,b,p,s}$ and $x_{o,b,p,s,r,w}$ variables. Constraints (5) concern the maximum weight that each train can bear and define the relation between $x_{o,b,p,s,r,w}$ and z_r variables. Constraints (6) and (7) impose that boxes assigned to wagons must be compatible with the wagon length and the weight limitations for each wagon. Constraints (8) regard the maximum handling operations that can be performed for each train at a given terminal.

4 Experimental Results

We coded the preprocessing analysis and the optimization procedure in C♯ using Cplex 11.0 as 0/1 LP solver. Then we ran some experiments to evaluate the performance of the

proposed planning approach, after having introduced in the statement of Problem 1 the possibility of not assigning all the boxes. To do this, we transformed equality constraints (3) into not greater or equal to inequalities and we added in the cost function the penalty term

$$M \cdot \sum_{o \in \mathcal{O}} \sum_{b=1}^{Nb_o} \left(1 - \sum_{p=1}^{Np_o} \sum_{s=1}^{Ns_{o,p}} y_{o,b,p,s} \right)$$

in order to minimize the number of possibly not assigned boxes (here M is a constant much larger than any other constant in the objective function).

We analysed two different scenarios for the network and trains, one called *Small* scenario (7 nodes, 20 links and 56 trains) and the other called *Large* scenario (10 nodes, 34 links and 98 trains). The considered planning horizon is one week, 2 types of wagons and 13 typologies of boxes are taken into account, each train covers either 2 or 3 links and has a number of wagons between 6 and 10. Data about orders were randomly generated: in particular, we generated 6 groups of 5 instances, called *SmallA* (20 orders and 3÷5 boxes per order), *SmallB* (30 orders and 4÷5 boxes per order), *SmallC* (40 orders and 5÷7 boxes per order) for the small scenario, whereas *LargeA* (40 orders and 2÷4 boxes per order), *LargeB* (50 orders and 4÷6 boxes per order), *LargeC* (60 orders and 6÷8 boxes per order) for the large scenario. We applied the preprocessing computation and, then, we solved these instances imposing the time limit of 2h for the Cplex solver. The tests were executed on a 2.8 GHz Pentium 4 computer with 2 GB of RAM.

In Table 1 we report the number of variables of the 0/1 LP formulation, the CPU time needed to solve the instance (in seconds), the percentage optimality gap, the number of not assigned boxes in the solution and the number of boxes that the solver proved that cannot be assigned in the optimal solution. We only show the computation time of Cplex solver because the computation time for preprocessing and model building can be considered irrelevant as it was always lower than 30 seconds. Note that, having penalized in the objective function the choice of not serving boxes, the last two columns in Table 1 are obtained as

$$NotAssSol = \left\lfloor \frac{Objective}{M} \right\rfloor \qquad NotAssProv = \left\lfloor \frac{LowerBound}{M} \right\rfloor$$

where $LowerBound$ denotes the value of the best lower bound found by the solver. We computed the optimality gap as

$$OptimGap = \frac{(Objective \mod M) - (LowerBound \mod M)}{Objective \mod M} \cdot 100$$

where the operator $\mod$ finds the remainder of the integer division between two numbers. In this way $Objective \mod M = Objective$ and $LowerBound \mod M = LowerBound$, when $NotAssSol = 0$ and $NotAssProv = 0$, but when not all the boxes are assigned, i.e., $NotAssSol \neq 0$ and $NotAssProv \neq 0$, the gap is computed by considering the objective function and the lower bound without the penalization terms.

Analysing the results in Table 1, we can note that the instances corresponding to the small scenario are solved in a satisfactory way, showing increasing difficulty form

Table 1. Cplex performances.

Instance ID	No. of variables	CPU time	$OptimGap$	$NotAssSol$	$NotAssProv$
SmallA1	16376	2216	*opt.*	0	0
SmallA2	25865	7200	7.83	0	0
SmallA3	25657	7200	0.17	0	0
SmallA4	32048	7200	2.85	0	0
SmallA5	19932	7200	0.02	0	0
SmallB1	36444	7200	5.77	0	0
SmallB2	41232	7200	3.28	0	0
SmallB3	47097	7200	5.40	0	0
SmallB4	44067	7200	2.57	0	0
SmallB5	36086	7200	5.76	0	0
SmallC1	55573	7200	3.06	11	10
SmallC2	72174	7200	4.76	12	11
SmallC3	81883	7200	3.55	5	0
SmallC4	74098	7200	8.77	0	0
SmallC5	63387	7200	3.69	7	7
LargeA1	35127	7200	5.66	0	0
LargeA2	44353	7200	11.98	0	0
LargeA3	29663	7200	2.54	0	0
LargeA4	43084	7200	5.27	0	0
LargeA5	43224	7200	5.00	0	0
LargeB1	83596	7200	11.48	0	0
LargeB2	95004	7200	10.22	0	0
LargeB3	84891	7200	-0.02	2	0
LargeB4	72837	7200	10.13	0	0
LargeB5	94950	7200	9.00	0	0
LargeC1	133251	7200	9.78	3	1
LargeC2	136996	7200	54.54	9	3
LargeC3	163414	7200	$+\infty$	424	0
LargeC4	136600	7200	11.21	19	0
LargeC5	140435	7200	8.04	12	2

Table 2. Cplex performances for group *LargeC* with time limit of 5 hours.

Instance ID	CPU time	$OptimGap$	$NotAssSol$	$NotAssProv$
LargeC1	18000	9.78	3	1
LargeC2	18000	54.54	9	3
LargeC3	18000	6.00	6	0
LargeC4	18000	6.87	2	0
LargeC5	18000	8.04	12	2

group *SmallA* to group *SmallC* (where in 4 over 5 instances some boxes are not assigned in the solution). The instances of the large scenario appear more difficult; in the groups *LargeA* and *LargeB* all boxes are assigned, except for instance *LargeB3* (in this instance the negative gap, that should be impossible by definition of the lower bound, must be considered as a proof of one more box impossible to serve, then it loses its usual

meaning). Finally, the results for the instances of group *LargeC* are not very satisfactory. We also solved the instances of group *LargeC* by imposing a time limit for Cplex of 5 hours (as shown in Table 2). For some instances we obtained a significant improvement, whereas for some other instances we found the same results produced after 2 hours of computation.

The groups of instances which can be considered more representative of a real case are groups *LargeB* and *LargeC*. They correspond to a network with 10 terminals and 34 links and to an average request of moving 250 and 420 boxes, respectively. These data can be considered quite realistic, at least for the first implementations in Italy of this innovative railway network that now is not yet applied. Moreover, we consider a time horizon of a week and, again, we think the choice of a weekly planning can make sense in a real application. These computational results make us think that the proposed planning procedure could be applied to a real system if the planning is realized off-line one or two days before the considered horizon. In this case, a much higher time limit (than 5 hours) could be set for the solver, thus probably yielding better solutions. Instead, if the solutions are needed in a short time and larger instances must be considered, it is necessary to develop different approaches (i.e. heuristic techniques) for addressing this planning problem.

5 Conclusions

In this paper we propose a planning procedure in order to meet transportation requests by using the railway network. The considered railway system is innovative because the terminals are supposed to be equipped with fast and automatic handling systems, allowing to make containers change different trains on their path from origin to destination. The solution to this planning problem has been divided in two phases: first the preprocessing analysis and, second, the optimization (solution of a 0/1 LP problem). To evaluate the effectiveness of the proposed planning approach we performed some experimental tests using a set of 30 random generated instances characterized by different dimensions.

The results obtained from these preliminary tests make us think that the proposed approach could be applied to a real system, but further extensions and investigations are needed as well. In fact, since the experimental tests were not completely satisfactory for the largest instances, some further heuristic techniques (e.g., based on relaxation or decomposition) are needed to speed-up and simplify the problem solution. These aspects will be considered in the future development of our research.

References

1. Crainic, T.G., Kim, K.H.: Intermodal transportation. In: Barnhart, C., Laporte, G. (eds.): Transportation. North Holland (2007)
2. Macharis, C., Bontekoning, Y.M.: Opportunities for OR in intermodal freight transport research: A review. European Journal of Operational Research, 153 (2004) 400–416
3. Wieberneit, N.: Service network design for freight transportation: a review. OR Spectrum, 30 (2008) 77–112

4. Bostel, N., Dejax, P.: Models and algorithms for container allocation problems on trains in a rapid transshipment shunting yard. Transportation Science, 32 (1998) 370–379
5. Corry, P., Kozan, E.: An assignment model for dynamic load planning of intermodal trains. Computers and Operations Research, 33 (2006) 1–17
6. Cordeau, J.-F., Toth, P., Vigo, D.: A survey of optimization models for train routing and scheduling. Transportation Science, 32 (1998) 380-404
7. Newman, A.M., Yano, C.A.: Scheduling direct and indirect trains and containers in an intermodal setting. Transportation Science, 34 (2000) 256–270
8. Sacone, S., Siri, S.: A planning approach for freight transportation operations in railway networks. Proc. of the IEEE International Conference on Automation and Logistics (2008)

A Risk Assessment Application of a Real Time Decision Support System Model for HAZMAT Transportation in a Sustainable Oriented Motorway Environment

Giampaolo Centrone[1], Maria Pia Fanti[2]
Gabriella Stecco[1] and Walter Ukovich[1]

[1] Dipartimento di Elettrotecnica, Elettronica ed Informatica
Università degli Studi di Trieste, Italy
{giampaolo.centrone,gstecco,ukovich}@units.it
[2] Dipartimento di Elettrotecnica ed Elettronica
Politecnico di Bari, Italy
fanti@deemail.poliba.it

Abstract. The transportation of hazardous material on congested motorways is an area of increasing concern for public safety and environmental awareness. This paper aims at developing a methodology with an original approach in making an attempt to encompass both professional experience and theoretical knowledge with application oriented studies from disparate areas related to the commercial transportation of HAZMAT on motorway, intimately linked with the "sustainable transportation" paradigm. The main objective is to assess quantitatively the acceptability of the Individual and Societal Risks connected with the transportation of HAZMATs. In addition, we propose a real time model of a Decision Support System for HAZMAT transportation on a sustainable oriented motorway environment. Finally, we offer an application of the proposed model. The case study involves a stretch of A4 motorway in the North-East of Italy.

1 Introduction

Economic globalization favors the increase of geographic mobility involving the expansion of transportation systems that is joined with the rise in land prices and the increase of air and noise pollution. In this world development [13], dangerous goods are used in many processes in industry all over the world and this has been justified by the economic revenue generated by their use. A dangerous good (named hazardous material or HAZMAT almost exclusively in the United States) is any solid, liquid, or gas that can harm people, other living organisms, property, or the environment. Due to its nature, every production, storage and transportation activity related to the use of HAZMAT have many risks for both society and environment and are often subject to chemical regulations. In this scenario, a new factor has acquired more and more importance: sustainability. Sustainability [5] is a systemic concept that relates to the continuity of economic, social, institutional and environmental aspects of human society. As HAZMATs are transported throughout the world in a great number of road shipments, their commercial transportation could be catastrophic and poses risks to life, health,

property, and the environment due to the possibility of an unintentional release. Transportation of HAZMATs on road actually represents a potentially high risk in regard to the nature of the HAZMAT carried by trucks and the physiochemical events associated with these materials (radioactivity, explosion, toxicity, corrosion etc.), the localization and the density of the concerned (population, economic activities, buildings, networks, infrastructures, natural areas etc.), the characteristics and state of the roads (topography, layout, tunnels etc.), the density of the traffic, and the environmental conditions (weather, natural events etc.). While HAZMAT accidents are rare events, in a sustainable vision of development it is necessary to integrate risk mitigation and prevention measures into the transportation management in order to avoid the risks turning into real events. In spite of this issue, HAZMAT type, quantity, itinerary and delivery time are not precisely known by the public authorities, the highway and motorway companies, and the population. As a consequence, one of the main objectives of research in this field is to provide appropriate answers to the safety management of HAZMAT shipments, in collaboration with the principal parties involved in the goods transportation process. Researches in this area [6], focuses on two main issues: i) to assess the risk induced on the population by HAZMAT vehicles traveling on the road network; ii) to involve the selection of the safest routes to take.

1.1 Problem Definition

In Italy about 80% of road traffic is represented by the delivery of goods, and the overall trend in Europe seems to predict an increase of 30% within 2010. About 18% of this freight traffic is currently represented by HAZMAT transportation, but a clear awareness of HAZMAT transportation world flows on road and on the other transportation modes - as well as of the related security and safety aspects - is not present yet, at least from a social and economic point of view. Intelligent Transportation System technologies have also made possible the gradual reduction in journey times and thus opening up new economic horizons, with the conquest of wider markets. The freedom gained by the ease of movement, however, has a cost in terms of environmental impact, quality of life and safety. The risk is that the increasing demand for current and especially future can make that the cost is no longer sustainable. However, the actual accident risk and impact is not calculated. In addition, when, due to unforeseen events (traffic jams, accidents, etc.), they need to change route, they do not have any particular guidance on the safest alternative route. Motorways are one of the most important supporting infrastructures of transportation networks: they assure efficient and safe mobility of persons and goods in the world and represent the largest part of the built environment. Motorway is a term for both a type of road and a classification or designation. Motorways are *high capacity roads* designed to carry fast motor traffic safely. In the E.U. they are predominantly dual-carriageway roads with a minimum of two lanes in each direction and all have grade-separated access. Motorways are comparable with North American freeways as road type, and interstates as classification. In Italy, according to [3], HAZMATs transportation by road should require constant monitoring (tracking and tracing) of vehicles and cargo handled. This requirement involves a series of obligations to which must meet companies under the European Agreement concerning the International Carriage of Dangerous Goods by Road (ADR). As a consequence, motorway concessionaires

must adopt real time systems to monitor HAZMATs carried and support the decisions on the transportation (MAS Monitoring - Alarm - Alerts).

2 Problem Solution

In Fig. 1 we propose a real time decision support system (DSS) model for monitoring of HAZMAT vehicles, aiming at solving the above stated problems.

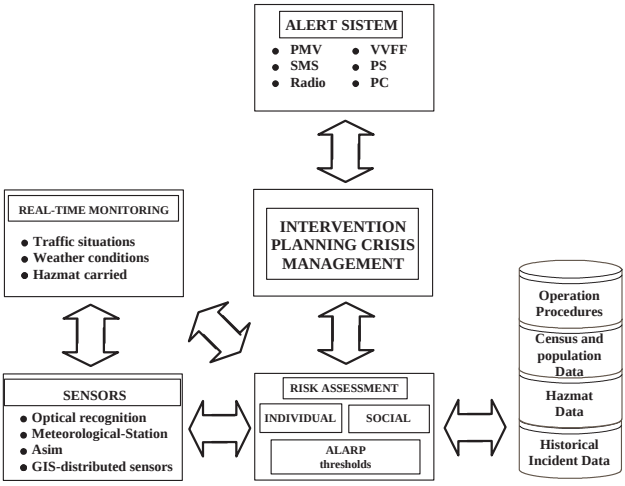
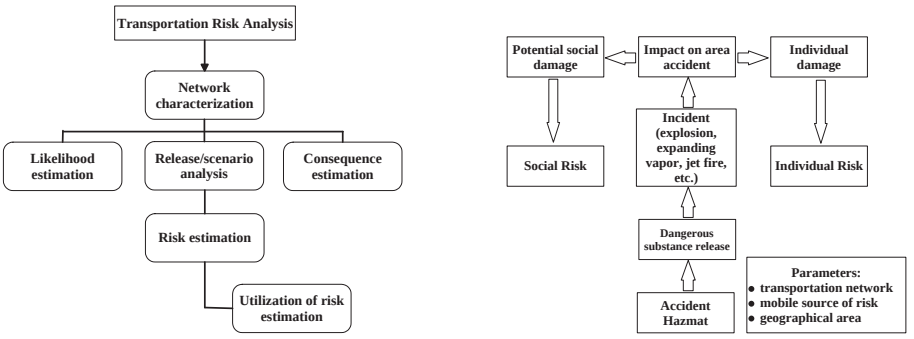


Fig. 1. The proposed System Architecture.

Such system should aim at calculating and evaluating in real time the individual and societal risk related to the transit of HAZMAT on the motorway network. Then, it should allow a monitoring in real time of the means transporting HAZMAT, a risk assessment derived from the carriage, the alert and notification of emergencies, and an anomalies reporting for a subsequent planned intervention. The model derives from the application of the quantitative risk assessment (QRA) methodology presented in Fig. 2(a).

We must take into consideration the following cause-effect chain which can be associated to a vehicle transporting one or more HAZMATs: the vehicle may be subject to a road accident (*accident*); the accident may cause the release of material transported (*release*); the release may cause a series of events (*incident*); the incident has an effect in the area surrounding the point accident. The model refers to damage to persons and in particular to death. The model of risk assessment derived from road transportation of HAZMAT is presented by a schematic representation in Fig. 2(b).

Risk assessment is typically structured as a process resulting from the interaction among the transportation network (in this case motorway), the vehicle (or better the traveling risk source), and the impact area. The model evaluates simultaneously the consequences and the frequencies of occurrence of possible scenarios. This makes it possible to assess



(a) The Transport Risk Assessment Methodology.

(b) The Risk Assessment Model.

Fig. 2.

quantitatively the *individual risk* and the *societal risk* (if the distribution of the people liable to be exposed is at hand). A complete assessment of the risks due to HAZMAT by road would require to consider all the possible weather situations, all the kinds of accidents, with all the types of vehicle partially or fully loaded. Such an evaluation is completely impossible and some simplifications have to be introduced. The QRA model is based on the following steps: choice of a restricted number of HAZMATs, choice of some representative accidental scenarios implying those HAZMAT with their usual packagings, identification of physical effects of those scenarios for an open air or a tunnel section, evaluation of their physiological effects on road users and local population, taking into account the possibilities of escape/sheltering, determination of the yearly frequency of occurrence for each scenario.
$F(N)$ curves and their expected values are the major outputs of the QRA model. According to [7], *Frequencies / Gravity curves* ($F(N)$ curves) stand for the annual frequency of occurrence F to have a scenario likely to cause an effect (generally, the number of fatalities) greater than or equal to N and *Expected value* (EV) is the number of fatalities per year, obtained by integration of a $F(N)$ curve.

3 How to Characterize the Risk in Transporting of HAZMAT on Motorway Environment: A Case Study

In this section we present an application of the real time model for the calculation of the Individual and Societal Risks in a motorway environment.

The Transportation Network. We consider the motorway network of an Italian concessionaire, in the North-East of Italy, S.p.A. Autovie Venete. In particular we refer to the sections summarized in Tab. 1. The network under consideration has been modeled as a network with nodes and links where nodes represent exits / junctions of the motorway and links the stretches (sections) of motorway between two exits / junctions. Let N_{link} be the set of links of the motorway network and l a generic link. For each link,

Table 1. Link length, average population density, HAZMAT and number of vehicles carrying them on tested links.

Link	Length (Km)	Average Population Density ($inhab/Km^2$)	Substance type	Num. of Vehicles
S.Stino di Livenza - Portogruaro	12.8	214.35	Chlorine	2
Portogruaro - Latisana	13.5	166.05	Ammonia	2
Latisana - S. Giorgio di Nogaro	17.6	112.66	Hydrochloric Acid	2
			Nitric Acid	2

the following data have been obtained: length [Km], and average population density [inhabitants / Km], around the link. According to [18], the average population density has been calculated using a GIS (Geographical Information System) overlapping the geographical map of the municipalities on the motorway network of S.p.A. Autovie Venete in order to identify common cross on each link. Then, for each link, we have identified the municipalities involved and measured the kilometers of infrastructure that pass through each town in order to identify the weights for calculating the average density on the link. These weights have been derived by dividing the kilometers of infrastructure that affect each municipality with the total length of the link. Note the density of population in each Italian municipality, using data on the census of 2001 [9], we shall calculate the weighted average with weights determined in the previous step. Tab. 1 illustrates the links in discussion with the relevant data.

The Accident Probability. We use the Truck Accident Rate of Harwood [8] in order to calculate the accident probability ($\lambda_{inc}(l)$) in terms of $events/(vehicle * km)$. We can also calculate the rate of accidents on a single stretch of length unit road by using the number of accidents in a time period of ten years and the total distance traveled by heavy vehicles during the same period, data provided by AISCAT (Associazione Italiana Societá Concessionarie Autostrade e Trafori) [1].

$$TAR_{y_r} = \frac{A_{y_r}}{VKT_{y_r}} \tag{1}$$

where TAR_{y_r} is the average accident rate for trucks $events/(vehicle*km)$ on the Italian motorway network for year y_r; A_{y_r} is the number of accidents involving trucks on the Italian motorway network; VKT_{y_r} is the total distance traveled (vehicle-kilometers) by trucks on the network under consideration. Tab. 2 shows the number of accidents involving heavy vehicles, the total distance traveled and the *Truck accident rate* year by year from 1997 to 2007 and the summary data (extension, routes) for the years under consideration. The last row presents the data that we use in the model.

The HAZMAT. in the next step, we select the HAZMATs that will be considered in calculating the risk. In particular, according to [16], we have considered substances that are more frequent or significant on the network under consideration. For each of these goods we obtained the following data (summarized in Tab. 3):

– the probability of release due to the accident ($p_{rel}(\nu)$) depending in general on the characteristics of the vehicle transporting the HAZMAT and on the type of accident where the vehicle is involved. This probability is taken from [2].

Table 2. Accidents and summary data year by year from 1997 to 2007.

		ROUTES	ACCIDENTS	TAR
YEAR	EXTENSION	$(veh.-km)$		
	KM	*Heavy*	*Heavy*	*Heavy*
1997	5371	$1.4428{*}10^{7}$	7825	$5.42{*}10^{-7}$
1998	5380	$1.5161{*}10^{7}$	8854	$5.84{*}10^{-7}$
1999	5380	$1.5974{*}10^{7}$	10024	$6.28{*}10^{-7}$
2000	5380	$1.6790{*}10^{7}$	9681	$5.77{*}10^{-7}$
2001	5388	$1.7254{*}10^{7}$	9647	$5.59{*}10^{-7}$
2002	5388	$1.7836{*}10^{7}$	9691	$5.43{*}10^{-7}$
2003	5388	$1.8359{*}10^{7}$	9198	$5.01{*}10^{-7}$
2004	5391	$1.9059{*}10^{7}$	8841	$4.64{*}10^{-7}$
2005	5432	$1.9184{*}10^{7}$	9005	$4.69{*}10^{-7}$
2006	5441	$1.9764{*}10^{7}$	9000	$4.55{*}10^{-7}$
2007	5446	$2.0229{*}10^{7}$	8613	$4.26{*}10^{-7}$
		$1.9403{*}10^{8}$	100379	$5.17{*}10^{-7}$

- the types of possible releases classified in relation to the size of the leakage hole ($N_{rel,t}(\nu)$) and the rate of release or the amount of material spilled ($p_{rel,t}$).
- the types of consequences of *incident* caused by different types of release of HAZMAT given the *accident* for a given type of substance ($N_{out}(\nu, r)$).
- the likelihood of occurrence of a final result given the *incident* ($p_{out}(i)$). This probability is derived for each triplet [substance - leakage - type of final outcome] from the information relating to incidents involving HAZMAT from 1997 to 2008 reported in the HMIS database [14]. This database contains detailed information on accidents involving HAZMAT in the U.S..
- the frequency of occurrence of a given scenario: $f^{scen}_{\nu,t}(i,r) = \lambda_{inc} \cdot p_{rel}(\nu) \cdot p_{rel,t}(r) \cdot p_{out}(i)$
- the lethal area radius of each pair [type of release - final outcome] calculated using the free software RMPComp distributed by U.S. EPA (Environmental Protection Agency) [15].

Assumptions. In the application we have considered the following assumptions.

1. $\lambda_{inc}(l)$ uniform throughout the link and constant for all links taken into consideration: λ_{inc};
2. exposure area of *danger circle* type [7] centered at the point of the incident with a radius depending on the type of substance, release and final outcome;
3. any person within the exposure area suffers from the same injury (death) in the same way regardless of the position, while people outside that area are not affected;
4. the seasons (j), the weather conditions on link $C^t_{met}(l)$ and the wind direction $\vartheta^t(l)$ are not taken into account;
5. the simulation is performed on a single moment in time;
6. risk neutral model ($\alpha = 1$) [17].

Individual Risk Calculation. The simulation was carried out on three adjacent links. As individual risk is the annual probability of an individual placed in a designated point

Table 3. Frequency scenarios.

HAZMAT	p_{rel}	**Release Type (Spillage)**	p_{relt}	**Incident Type (Cloud)**	p_{out}	**Incident Prob.**	**Scenario Frequency**	**Lethal area radius (Km)**
Chlorine	0.010	Small	0.94	Toxic	1	$9.40*10^{-3}$	$4.86*10^{-9}$	1.0
	0.010	Medium	0.04	Toxic	1	$4.00*10^{-4}$	$2.07*10^{-10}$	2.8
	0.010	Large	0.02	Toxic	1	$2.00*10^{-4}$	$1.03*10^{-10}$	5.6
Ammonia	0.025	Small	0.93	Toxic	1	$2.31*10^{-2}$	$1.20*10^{-8}$	0.2
	0.025	Medium	0.05	Toxic	1	$1.33*10^{-3}$	$6.88*10^{-10}$	1.0
	0.025	Large	0.02	Toxic	1	$5.32*10^{-4}$	$2.75*10^{-10}$	2.1
Nitric Acid	0.015	Small	0.93	Toxic	1	$1.39*10^{-2}$	$7.19*10^{-9}$	0.3
	0.015	Medium	0.06	Toxic	1	$8.82*10^{-4}$	$4.56*10^{-10}$	0.5
	0.015	Large	0.01	Toxic	1	$2.21*10^{-4}$	$1.14*10^{-10}$	1.9
Hydrochloric Acid	0.015	Small	0.92	Toxic	1	$1.38*10^{-2}$	$7.13*10^{-9}$	0.3
	0.015	Medium	0.05	Toxic	1	$7.37*10^{-4}$	$3.81*10^{-10}$	0.8
	0.015	Large	0.03	Toxic	1	$4.81*10^{-4}$	$2.49*10^{-10}$	2.6

Table 4. Geographical coordinates of the points chosen for Individual Risk calculation.

Point Location	**Latitude**	**Longitude**
Portogruaro Centro	45.78	12.83
Area di Servizio Fratta Nord	45.80	12.88
Latisana Ospedale	45.77	13.00
Muzzana del Turgnano Centro	45.82	13.13

of interest is affected by some degree of damage as a result of a specific incident [10], four points were chosen as "hot spots" at which to calculate the individual risk. Tab. 1 and 4 show respectively the links with the relevant substances circulating and the geographical coordinates of the points chosen for the calculation of individual risk. The network portion and the points under consideration are represented in Fig. 3.
For the calculations we have used the formula presented in [11] and [12] suitably modified to take into account the previous assumptions, the motorway environment and the real time events. More details can be found in [4]. Consequently, we made explicit that each event i belongs to the $N_{(out)}(\nu)$ of the general model may consist of a pair [type of release - final outcome] as in this case.

$$\mathrm{IRP} = \sum_{i=1}^{N_{links}} \sum_{v=1}^{N_{veh}(l)} \sum_{r=1}^{N_{rel}(\nu)} N_{type}(l,\nu) f_{rel}(\nu,r) \cdot \cdot \int_{L_l} \sum_{i=1}^{N_{out}(\nu,r)} p_{out}(i) \cdot V_{Q(x)v \to S}(i) dL_l \quad (2)$$

$$f_{rel}(\nu,r) = \lambda_{inc} \cdot p_{rel}(\nu) \cdot p_{rel,t}(r) \quad (3)$$

Fig. 3. Representation of points and the network portion under consideration.

Table 5. Simulation results - Individual Risk.

Point Location	Individual Risk
Portogruaro Centro	$2.28{*}10^{-9}$
Area di Servizio Fratta Nord	$5.78{*}10^{-9}$
Latisana Ospedale	0
Muzzana del Turgnano Centro	$1.75{*}10^{-9}$

where $N_{veh}(l)$ is the number of different vehicle topologies on link l, $N_{type}(l,\nu)$ is the number of vehicles carrying the dangerous substance ν currently in transit on the link l, $N_{rel}(\nu)$ is the number of release cases of the dangerous substance ν, L_l is the route of link l and $V_{Q(x)_\nu \to S}(i)$ is equal to 1 if the point S is inside the *danger circle* centered at the point of possible accident Q related the triplet [substance - release type - final outcome]; 0 if the point S is external the same *danger circle*. Line integral was calculated using the method of Cavalieri-Simpson, dividing each of the three links in 10 intervals of equal length. Tab. 5 shows the results of the simulation.

From the evidence we can establish that the individual risk in the four points is acceptable according to the British ALARP threshold as the value is much lower than the limit value of 10^{-6} [10].

Societal Risk Calculation. In order to calculate the societal risk, we refer again to [11] and [12] suitably modified. For each link knowing the vehicles that are going through, we use (4) to obtain the $F(N)$ curves representation (for details see [4]).

$$F(N) = \sum_{i=1}^{N_{link}} \sum_{v=1}^{N_{veh}(l)} N_{type}(l,\nu) \cdot \sum_{i=1}^{N_{out}(\nu)} \int_{L_l} \delta^N_{scen}(i, C_{met}(l)) dL_l \qquad (4)$$

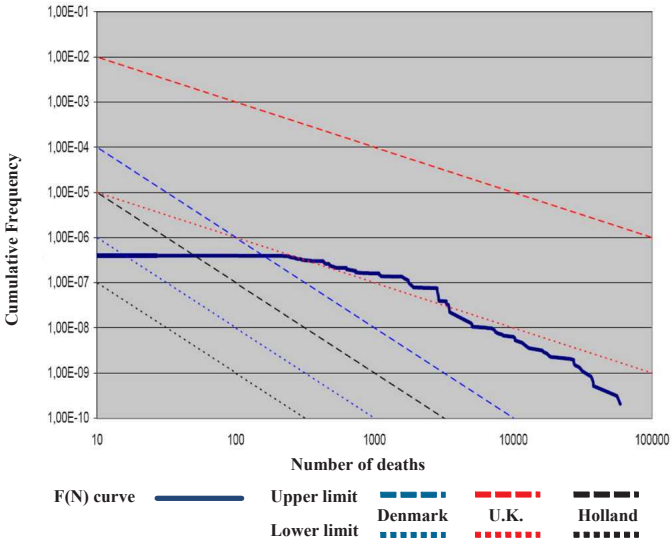


Fig. 4. Simulation results - Societal Risk.

The curve $F(N)$ is drawn in Fig. 4, referred to the simulation with different thresholds of acceptability [12].

It can be seen that at the moment of the simulation the societal risk, according to the British acceptability thresholds, is in the ALARP zone, whereas according to Dutch and Danish thresholds it is not acceptable.

4 Conclusions

The transportation of HAZMAT on congested motorways is becoming an area of increasing concern for public safety and environmental awareness. The risk to population and damage to environment is a major concern to the general public and government policy makers. Against these problems we present a methodology to perform the individual and societal risk assessment related to HAZMAT transportation in a sustainable oriented motorway environment. It constitutes an approach based on the GIS. The assessment criteria, based on the "sustainable transportation" paradigm, are structured into efficiency, cohesion and environmental criteria. The aim is assessing whether these risks are acceptable and possibly, if they were not, notify the situation through alert messages in order to take appropriate actions. We offered an application of the proposed real time model for the calculation of the Individual and Societal Risks involving in the case study a stretch of A4 motorway in the North-East of Italy. In spite of a limited number of trucks transporting HAZMAT on the motorway, the results of the application point out the concrete possibility to exceed the thresholds of the ALARP limits for the societal risk.

With regard to possible developments, the QRA methodology could be to extend, in

particular the model for calculating the individual and societal risk, to other situations of HAZMAT transportation by other transportation modes.

References

1. AISCAT: Rapporto incidentalitá rete autostradale italiana 1997-2007. Statistiche Aiscat, Associazione Italiana Societá Concessionarie Autostrade e Trafori, (2008)
2. Brown, D.F., Dunn, W.E.: Application of a Quantitative Risk Assessment Method to Emergency Response Planning. Computers & Operations Research, 34 (2007) 1243–1265
3. CEI: Road traffic and transport telematics Standards framework. Review of the applications and technological trends, normative references Annex to CEI UNI 70031 (2001) 1–18
4. Centrone, G.: Modeling a real time decision support system for HAZMAT transportation in a sustainable oriented motorway environment. Ph.D. Thesis, Universitá di Trieste (2009)
5. Centrone, G., Coslovich, L., Pesenti, R., Ukovich, W.: Social responsibility and sustainability in motorway Corporate Governance. International Journal of Environment and Sustainable Development 7 (2008) 94–113
6. Centrone, G., Pesenti, R., Ukovich, W.: Hazardous Materials Transportation: a Literature Review and an Annotated Bibliography. In: Bersani, C., Boulmakoul, A., Garbolino, E., Sacile, R. (eds.): Advanced Technologies and Methodologies for Risk Management in the Global Transport of Dangerous Good. NATO Science for Peace and Security Series, Vol. 45. IOS Press, Amsterdam (2008) 33–62
7. Erkut, E., Tjandra, S.A., Verter, V.: Hazardous materials transportation. In: Barnhart, C., Laporte, G. (eds.): Transportation, Handbook in OR & MS, Vol 14. Elsevier (2007) 539–621
8. Harwood, D.W., Russell, E.R., Viner, J.G.: Procedure for developing truck accident and release rates for hazmat routing. Journal of Transportation Engineering 119 (1993) 189–199
9. ISTAT: Quattordicesimo censimento generale della popolazione italiana (2001). http://cartema2.istat.it/home_cartema/iisstart.htmh
10. Jonkman, S.N., van Gelder, P.H.A.J.M., Vrijling, J.K.: An overview of quantitative risk measures for loss of life and economic damage. Journal of Hazardous Materials 99 (2003) 1–30
11. Leonelli, P., G. Spadoni, G.: A new numerical procedure for calculating societal risk from road transport of dangerous substances. In Proceedings SRAEurope Annual Meeting (1996) 243–246
12. Leonelli, P., Bonvicini, S., Spadoni, G.: Hazardous materials transportation: a Risk-analysis-based routing methodology. Journal of Hazardous Materials 71 (2000) 283–300
13. Nederveen, A.A.J., Konings, J.W., Stoop, J.A.: Globalization, International Transport And The Global Environment: Technological Innovation, Policy Making And The Reduction Of Transportation Emissions. Transportation Planning and Technology 26 (2003) 41–62
14. PHMSA - U.S. Department of Transport: HMIS database. (2006) https://hazmatonline.phmsa.dot.gov/IncidentReportsSearch/Search.aspx
15. RMP*Comp 1.07 http://www.epa.gov/OEM/content/rmp/rmp_comp.htm
16. Rossetto, A.: Sviluppo di un sistema DSS per il monitoraggio e l'analisi quantitativa del rischio nel trasporto di merci pericolose in autostrada. MSc. Thesis, Universitá di Trieste (2008)
17. Slovic, P.: Perception of risk. Science (1987) 280–285
18. Zhang, J.J., Hodgson, J., Erkut, E.: Using GIS to assess the risks of hazardous materials transport in networks European Journal of Operational Research 121 (2000) 316–329

Robust Navigation for an Autonomous Helicopter with Auxiliary Chattering-free Second Order Sliding Mode Control

S. Vite-Medécigo, Ernesto Olguín-Díaz and Vicente Parra-Vega

Robotics and Advanced Manufacturing
Research Center for Advanced Studies–CINVESTAV, Saltillo, Mexico
{silvionel.vite,ernesto.olguin,vicente.parra}@cinvestav.edu.mx

Abstract. This paper presents a novel technic for autonomous flight and navigation control of AAVs, particularly useful for helicopters. Three servo-loop controller are introduced to yield stable robust regulation. The inner control loop is based on an LQR regulator designed over the linearized plant at hover to guarantee close-loop stability. The middle loop is a feedback linearization controller based on the close-loop linearized system to cope with the underactuated nature of the helicopter, by guaranteing an asymptotically stable zero dynamics. Finally the outer control loop enforces a tracking second-order sliding-mode for cartesian position and heading navigation outputs. The simplicity of this control proposal allows easier and intuitive guidelines to tune feedback gains while the chattering-free sliding-mode fulfills basic robustness properties, ideal for this complex systems subject to external disturbances like wind gusts.

1 Introduction

Automatic flying vehicles, also known as Autonomous Aerial Vehicle (AAV), represents a huge field of applications in particular for advanced automatic control techniques because human intervention is considered difficult or dangerous. There are wide civil and military interests in helicopters, like traffic surveillance, air pollution monitoring, area mapping, agricultural applications, exploration, scientific data collection, search and rescue.

Among the AAVs, the rotary wing AAVs such as the helicopter has the advantage of having the ability to perform different flight regimes like hover, backward, lateral of pure vertical flight, in contrast to fix wing such as typical airplanes. However, helicopters are underactuated mechanisms whose dynamic model exhibits high nonlinearities with physical parameters hard to measure precisely. The operational versatility of helicopters requires complex controllers to achieve such flight regimes.

We can classify two type of controllers. One uses the full dynamic modeling with simple model-free controllers; the second assumes simple dynamic modeling used in complex controllers design. In the former case, due to the complexity of the full dynamic model of helicopters and unknown aerodynamic/aeroelastic parameters, model-based controllers are hard to implement and then simpler control laws based on linearized plant are preferred. Since this approach is prone to instability due to the un-

knowns of the dynamic plant, an auxiliary ν controller is added, typical PID-like controller, which introduces limited performance because of the well-known limitations of these PID-like controllers. However this type of studies has been useful to understand better the complexity and structural properties on real applications because they employ the full model with simple controllers providing clear intuitive understanding on the stability properties of the closed-loop system. The latter case uses simpler dynamic models, based on restrictive academic assumptions, such as the helicopter is constrained to move only in a subset of $\Re^6$, exhibiting pseudo-flying conditions with model-based controllers [4]. This approach guarantees very limited performance in real conditions, with limited scope of real applications.

In this paper, we focus our attention in the full dynamical model of the scalar R/C X-cell90 helicopter, and propose a novel auxiliary controller based on a chattering-free sliding modes, which increases the closed-loop performance because it is a tracking-designed controller with inherent robustness capabilities. This allows to guarantee better closed-loop performance in comparison to auxiliary controllers based on PID-like controllers. Simulations under external disturbances like wind gusts, wherein clearly verifies the validity of the proposed approach.

2 Relevant Background

Complex helicopter models, [9, 12], based in the Newtonian model of a free flying rigid objet are restricted to measurements on the center of mass, which indeed can vary in real conditions, neglecting at small velocities the Coriolis effects, thus this model is not useful in aggressive maneuvers or wide range of operational flight conditions. More over, dissipative effects on the fuselage are not taken in account that would be important during the navigation. In [6] this Coriolis effects are taken in account but simplifies the 6-DOF Inertia-Matrix to be completely diagonal. More over, a spring model is included to describe the main rotor forces mapping to the main body rigid object modeling. Nonetheless these models neglect the blade's kinetic energy, which can be up to 20 times the one of the fuselage [1]. Thus, in hover regime this energy must be taken in account to give rise to a dynamic model of more than the 6 degrees of freedom (DOF) of a rigid free flying object, showing the complexity of the main rotor itself. This model is more relevant in practice since it includes this important energy.

On one hand the forces acted in the rigid free flying object (the fuselage) are given mainly by the forces exerted at the main and tail rotors. The forces at the tail rotor is a simple thrust in the direction perpendicular to the tail rotor whose magnitude changes with the tail collective. On the other hand, the main rotor provides 3 Cartesian components of the main rotor thrust given by the main rotor collective and two azimuth angles also known as lateral and longitudinal cyclic. Then, even for the most simplest model, *i.e.* 6-DOF, the full system is underactuated because the control dimension is 4.

The problem of control design for this kind of systems even for complex models including all or some of the full main rotor dynamics as been addressed extensively in the literature, however the control of the underaction remains open, though it has been addressed in [2, 13]. In particular, [14] proposes LQR-BDU techniques a the linearized model, concluding a robust regulator in a small neighborhood of the linearized point.

LQR feedback control scheme plus an additional PID-like regulators loop is a popular choice because the unknown parameters and external disturbances, like gust of wind, deviates the operational point; however the popular integral-loop may increase the sensitivity of the system under commonly time-varying disturbances. In this paper, the additional servoloop is based on a robust chattering-free sliding mode controller to provide wider operational conditions, with better performance.

3 Mathematical Model

In contrast to the Lagrange method, the equations obtained via Newton's laws expressed with velocities and acceleration measured at the body (relative to the body's frame and not to the inertial one) result in a simpler representation. The difference in these representations arise from the fact that the generalized coordinates needed in the Lagrange method, while having a physical meaning in the pose, the generalized velocity does not have a physical meaning and neither the generalized force vector; at least part of them. Equivalences between these two different representations can be obtained via the kinematic equation, *i.e.* using the mapping operator that express the physical meaning of velocity wrench used in Newton formulation out of the generalized velocity vector used in Lagrange one [5, 10].

The kinematic of a rigid single body in space is represented only by the pose (position and attitude) of the body with respect to an inertial (fixed) frame Σ_0, where Σ_v is the frame rigidly attached to the object. See Fig. 1.

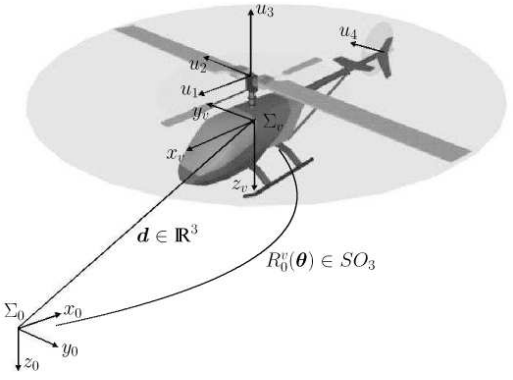


Fig. 1. Inercial frame Σ_0 and object frame Σ_v.

The rotation matrix $R_0^v \in SO_3$ transfers a 3D vector from its representation in frame Σ_v to the inertial frame Σ_0. The generalized position of the object, expressing both position and attitude of the object is then defined as

$$q \triangleq \begin{pmatrix} d \\ \theta_v \end{pmatrix} \in \Re^6 \tag{1}$$

where $d = (x, y, z)^T \in \Re^3$ is the object inertial position with respect to the Σ_0 given by the inertial Cartesian coordinates of the origin of frame Σ_v and $\theta_v =$

$(\phi_x, \theta_y, \psi_z)^T \in [-\pi, \pi] \times [-\pi/2, \pi/2] \times [-\pi, \pi]$ is the set of attitude parameters (in this case the roll-pith-yaw Euler angles) of Σ_v with respect to Σ_0. For this very set of attitude parameter the form of the rotation matrix R has a particular expression that can be found in either [5, 10]. The vector $\nu \in \Re^6$ is the velocity twist which defines the linear and angular velocity of Σ_v expressed in the non-inertial frame Σ_v, i.e. the velocity measured from the object

$$\nu \triangleq \begin{pmatrix} v \\ \omega \end{pmatrix} \in \Re^6 \tag{2}$$

where $v = R_0^{v^T} \dot{d} \in \Re^3$ is the lineal velocity of the object and $\omega \in \Re^3$ is the angular velocity of frame Σ_v, both vectors expressed in the non-inertial frame Σ_v. In strictly mathematical sense $R_0^{v^T} \dot{\theta}_v \neq \omega$, however there is a relationship given by $\omega = R_0^{v^T} J_\theta \dot{\theta}_v$, where $J_\theta \in \Re^{3\times 3}$ is a linear operator given by attitude parameters. Then a relationship between ν and $\dot{q}$ is found as follows

$$\nu = J_v(q)\dot{q} \tag{3}$$

with $J_v(q) \in \Re^{6\times 6}$ being the linear operator of the kinematic equation. The Kirchhoff formulation for the equation of motion of a rigid object is nothing but the moment conservation equations expressed in the non-inertial frame in terms of the kinetic energy as

$$\frac{d}{dt}\frac{\partial K}{\partial v} + \omega \times \frac{\partial K}{\partial v} = f \tag{4}$$

$$\frac{d}{dt}\frac{\partial K}{\partial \omega} + \omega \times \frac{\partial K}{\partial \omega} + v \times \frac{\partial K}{\partial v} = n \tag{5}$$

where f and n are the forces and torques respectively that acts over the object, including gravity, dissipative forces and any external input force acting on the object, and K is the kinetic energy as $K = \frac{1}{2}\nu^T M \nu$, where matrix $M \in \Re^{6\times 6}$ is the *Inertia Matrix* with respect to the origin of frame Σ_v, defined as follows:

$$M \triangleq \begin{bmatrix} mI_3 & -m[r_c\times] \\ m[r_c\times] & I_g \end{bmatrix} \tag{6}$$

which is by construction constant, positive definite and symmetric $M = M^T > 0$. The terms of this Inertia Matrix are the total mass m of the object, the distance from the origin of frame Σ_v to the center of mass of the body r_c, expressed in the body's frame, the inertia moment matrix I_g computed from the origin of Σ_v, and the skew symmetric matrix representation of the cross product $[a\times]b = a \times b$.

Equations (4)-(5), after proper algebraic manipulation and using the kinetic energy expression above, can also be expressed in a single vectorial equation as

$$M\dot{\nu} + c(\nu) = F,$$

where matrix $M \in \Re^{6\times 6}$ is the *Inertia Matrix* with respect to the origin of frame Σ_v, the vector $c(\nu)$ regroups all the nonlinear terms and is known as the Coriolis vector, and

$F \triangleq \left(f^T, n^T\right)^T = F_G + F_D + F_T$ is the force wrench consisting in gravity, dissipation and thrust wrenches respectively.

Because of the quadratic nature in terms of velocity Coriolis vector it can also be expressed as product of a matrix and the velocity wrench: $c(\nu) = C(\nu)\nu$. The matrix $C(\nu)$, referred as the Coriolis matrix may have many different representations, but at last one of them fulfills the skew-symmetry property $C(\nu) + C(\nu)^T = 0$.

F_G, being the gravity force wrench in the objects frame, can be computed rotating the gravity influence to the objects frame $f_g = mgR_0^{v^T}\boldsymbol{k}$. The gravity vector is defined then as $g(q) \triangleq \left(f_g^T; 0\right)^T$. Then $F_G = -g(q)$, where the negative sign comes from the fact that the positiveness of the vertical axis z_0 is pointing downward, to the center of the earth, due to convention in vessel engineering.

F_D are the dissipation aerodynamic forces and these are by nature quadratic and homogeneous to the velocity wrench. Then a possible approach to model these forces can be given as $F_D = -D\left(\|\nu\|\right)\nu$, where the damping matrix should be definite positive $D > 0$ to fulfill passivity [10].

Finally, F_T are thrust aerodynamical wrench and are given by the influences of the forces exserted by both rotors. There are 3 forces at the center of the main rotor given by longitudinal cyclic (u_1), the lateral cyclic (u_2) and the collective (u_3). There is also a fourth force at the center of the tail rotor (u_4) (See Figure 1). This mapping is given by a constant operator $B_e \in \Re^{6\times 4}$ that can be computed from the geometry of the rotors with respect to vehicle's frame Σ_v as $F_T = B_e u$, with $u = (u_1,\, u_2,\, u_3,\, u_4) \in \Re^4$ and B_e a column full rank matrix.

The dynamic modeling of the helicopter without considering the rotors dynamic is then given by [10]:

$$M\dot{\nu} + C\left(\nu\right)\nu + D\left(\|\nu\|\right)\nu + g\left(q\right) = B_e u \tag{7}$$

$$\nu = J_v(q)\dot{q} \tag{8}$$

which can be expressed is state space form using the state definition $x \triangleq (q^T, \nu^T)^T$.

4 Controller Design

A robust control law is necessary due to the environmental nature of AAV, then LQR approach is preferred because it is an optimal criteria for set-point control while minimizing energy consumption [8]. However this technic is based on a linear model or a linearized one, which means it works as supposed only in the operational point x_o, where the linearization was computed with $\tilde{x} = x - x_o$:

$$\dot{\tilde{x}} = A\tilde{x} + Bu \tag{9}$$

$$y = C\tilde{x} \tag{10}$$

In the case of the system (7)-(8) the state realization yields to

$$A(x) = \begin{bmatrix} \frac{\partial}{\partial q}\left(J_v^{-1}(q)\nu\right) & J_v^{-1}(q) \\ -M^{-1}\left(\frac{\partial}{\partial q}g(q)\right) & -M^{-1}\left[C(\nu) + D\left(\|\nu\|\right)\right] \end{bmatrix} \in \Re^{12\times 12} \tag{11}$$

$$B = \begin{bmatrix} 0 \\ M^{-1}B_e \end{bmatrix} \in \Re^{12\times 4}, \qquad C = \begin{bmatrix} I & 0 \end{bmatrix} \in \Re^{6\times 12} \tag{12}$$

Remark 1. Clearly, (12) indicates that $CB = [0] \in \Re^{6\times 4}$.

For the particular case where the operation point is *hover*, i.e. $x_o = \left(q_d^T;\ 0\right)$ and $q_d = (x_d, y_d, z_d, 0, 0, 0)^T$ the state matrix becomes constant:

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}\left[\frac{\partial}{\partial q} g(q)\right] & 0 \end{bmatrix} \in \Re^{12x12} \tag{13}$$

for the same pair (B, C). From (13) it can be seen that the linearized model at hover operational point has all the eigenvalues at the origin. This is due to the double integrator nature of the system and the fact that the aerodynamic dissipation forces are quadratic to the velocity which becomes null at the steady state. This explains the high degree of instability of such systems.

Remark 2. Notice that the pair (A, B) is controllable, then a linear state feedback ($u = -Kx$) would enforce a desired closed-loop system stability and performance at the operation state x_o [3].

Remark 3. The product $CAB = M^{-1}B_e \in \Re^{6\times 4}$ is column full rank constant matrix, and column full rank matrix elsewhere: $CA(x)B = J_v^{-1}(x_1)M^{-1}B_e \in \Re^{6\times 4}$.

4.1 Feedback Linearization

Stability of the equilibrium point x_o is only local and valid only in its very narrow neighborhood. When the dynamic model deviates or the system is subject to bounded unmodeled dynamics or bounded disturbances. To cope with that an auxiliary feedback control is commonly proposed [2],

$$u = -Kx + v \tag{14}$$

where K is computed via LQR feedback scheme and v is an additional auxiliary control input. Then, the linearized close-loop system can be written as

$$\dot{x} = [A - BK]\, x + Bv \tag{15}$$

$$\bar{y} = \bar{C}x \tag{16}$$

where $\bar{y}$ is only a part of the originally output ($y = q$), defined, as the Cartesian position and heading only, excluding the roll and pitch attitude angles: $\bar{y} \triangleq (x, y, z, \psi_z)^T$. The output matrix $\bar{C} = [C_1\ 0] \in \Re^{4\times 12}$ with $C_1 \in \Re^{4\times 6}$ has raw full rank. Notice that $\bar{C}B = [0] \in \Re^{4\times 4}$ still holds, consequently the first and second time derivatives of the new output become

$$\dot{\bar{y}} = \bar{C}Ax \tag{17}$$

$$\ddot{\bar{y}} = \bar{C}A\,[A - BK]\, x + \bar{C}ABv \tag{18}$$

Remark 4. Matrix $\bar{C}AB = C_1 M^{-1} B_e \in \Re^{4\times 4}$ is full-rank invertible matrix, thus stable zero dynamics arise, that is the roll and pitch attitude angles are stable, [7].

The Feedback Linearization controller (FL), issued from eq. (18) would have the form

$$v = \left[\bar{C}AB\right]^{-1} \left(\bar{v} - \bar{C}A\left[A - \mathbf{BK}\right]x\right), \tag{19}$$

yielding to a closed-loop system $\ddot{\bar{y}} = \bar{v}$, as reported in [2]. However this is rather awkward since the LQR state feedback $(-Kx)$ is canceled in (14) by this second loop. Since it is preferable to maintain an optimal stabilizable regulator such as the LQR in the control loop a Partial Feedback Linearization (PFL) is proposed as:

$$v = \left[\bar{C}AB\right]^{-1} \left(\bar{v} - \bar{C}A^2 x\right) \tag{20}$$

which delivers a second order coupled linearized close-loop system

$$\ddot{\bar{y}} = \bar{v} - \bar{C}ABKx \tag{21}$$

Notice that dynamics $-\bar{C}ABKx$ represents the a residual coupled dynamics introduced by the *optimal* LQR regulator and because of the underactuated nature of this system.

4.2 Sliding-Mode Control

Let $\Delta\bar{y} = \bar{y}_d - \bar{y}$ be the output tracking error, where y_d is the desired output signal, and choosing the new second order sliding-mode control law $\bar{v}$ given by

$$\bar{v} \triangleq \ddot{\bar{y}}_d - \alpha\dot{\Delta\bar{y}} + \beta s_0 e^{-\beta t} - K_i tanh(\sigma s_q) - K_d s_r \tag{22}$$

for large enough gains K_d, K_i and small error on initial conditions, with $s_r = s_q + K_i \int \mathrm{sgn}(s_q)$, $s_q = s - s_d$, $s = \dot{\Delta\bar{y}} + \alpha\Delta\bar{y}$ and $s_d = s_0 e^{-\beta t}$, $s_0 = s(t_0)$. The function $tanh(*)$ stands for a the sigmoid hyperbolic tangent function with $\sigma > 0$, not necessarily large. Then, the complete control law is given by

$$u = \left[\bar{C}AB\right]^{-1} \left[\ddot{\bar{y}}_d - \alpha\dot{\Delta\bar{y}} + \beta s_0 e^{-\beta t} - K_i tanh(\sigma s_q) - K_d s_r - \bar{C}A^2 x\right] - Kx \tag{23}$$

Substituting (23) into (9) yields

$$\dot{s}_r = -K_d s_r - \bar{C}ABKx - K_i Z \tag{24}$$

for bounded $Z = tanh(\sigma s_q) - sgn(s_q)$. Finally, we can state the main result.

Theorem 1. *Consider (23) into (9), then the closed loop (24) gives rise to robust exponentially stable dynamics of tracking errors, under a chattering-free second order sliding modes for all time, with stable zero dynamics.*

Proof. It follows closely [11], **QED**.

Remark 5. The state feedback stabilize locally the operation point, decouples the close-loop dynamics of the lateral, longitudinal, vertical, and heading navigation and preserves stability of the zero dynamics. Additionally, the auxiliary control input enables a wider operational region by adding robustness to the overall closed loop control.

5 Results

Consider the nonlinear model of an X-cell90 R/C helicopter. The linear model is computed, for simulation simplification, at the operating point $x_o = (0, 0, 0, 0, 0, 0)^T$. In Table 1 initial conditions and gain tuning for the output feedback sliding mode are shown. For comparison purposed, simulation using Matlab are also performed commuting the auxiliary control (14) for a properly tuned PD control.

Table 1. Initial conditions and tuning gains for the sliding-mode control.

Initial conditions & SMC-Gains				
	x	y	z	ψ
q_0	-2.1	1.05	0.11	0
$\dot{q}_0$	0	0	0	0
α	3.15	3.15	4.5	15
β	1	1	1	1
k_d	15.6	15.6	56.16	234
k_i	3.51	3.51	27.8	52.65

Figure 2 shows the 3D trajectory and the tracking error of both position and attitude for the helicopter when the control law is the two servo-loop, similar to the one presented in [2] (FL-PD), consisting in a Feedback Linearization (which also cancels de LQR inner loop) and a PD controller. As it can be seen this PD controller cannot reject constant disturbances as gravity. Figure 3 shows the same trajectory tracking with the proposed Sliding-Mode robust controller in the place of the PD above (FL-SM). This controller consist in a Feedback Linearization and a second order Sliding-Mode output feedback. It can be seen a good performance on the desired position tracking, including the heading (yaw angle), even in the presence of random disturbance forces (for gust of winds). The roll and pitch angles, which define the zero dynamics, are stable, which is in accordance with the feedback linearization design. Figure 4 shows also the trajectory tracking as in the previous Figures. The difference here is that in this case the middle loop does not cancel the LQR inner loop, and the residual dynamics are coped by the outer second order Sliding-Mode loop. This controller, given by (23), is called in this work as LQR-PFL-SM. Evident differences in the performance of the FL-PD and the FL-SM can be seen mainly because the PD cannot overcome constant disturbances as the gravity effect. Small differences between the FL-SM scheme and LQR-PFL-SM one can be seen at the magnitude level of the Cartesian position tracking error where are smaller in the second, because the Sliding mode acts since the initial conditions, tracking almost perfectly the desired trajectory. In attitude there are no significative differences founded.

6 Conclusions

Control of autonomous helicopters in the presence of environmental and system uncertainties is a challenging task. These uncertainties not only modify the dynamics be-

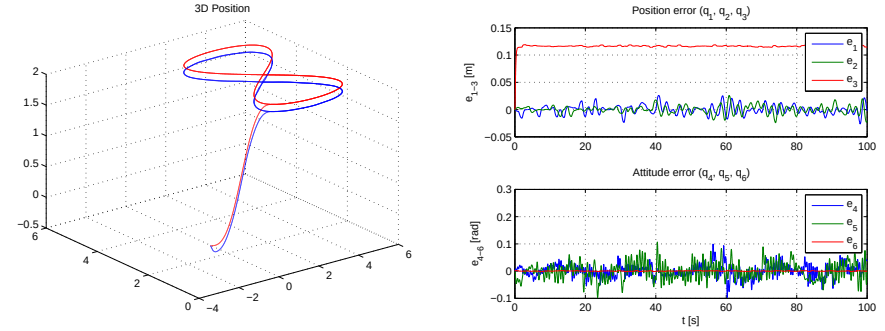


Fig. 2. Space position trajectory tracking in 3D and pose tracking errors for a FL-PD control law.

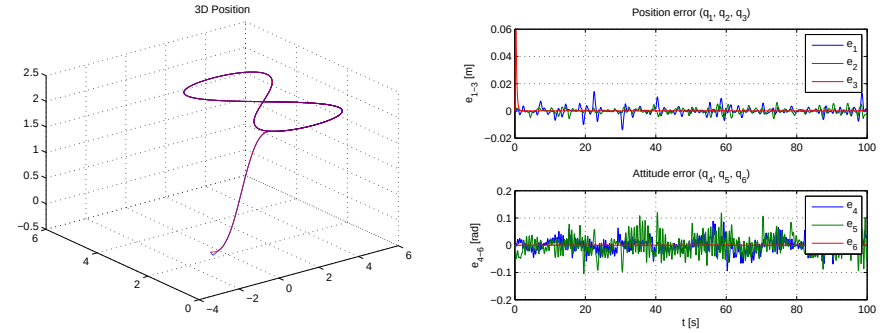


Fig. 3. Space position trajectory tracking in 3D and pose tracking errors for the FL-SM control law.

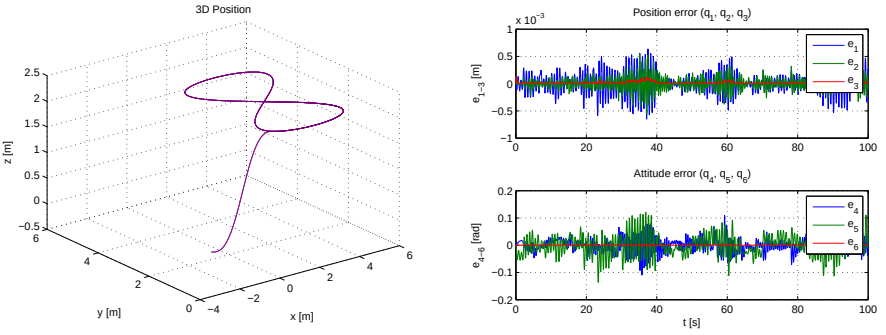


Fig. 4. Space position trajectory tracking in 3D and pose tracking errors for LQR-PFL-SM control law.

havior of the system, but also the trim inputs themselves. What is therefore needed is a viable controller capable of simultaneously accommodating all coupling features, parametric uncertainties, and trim errors. State representation is necessary to perform both tangent linearization for the design of an ideal Optimal stable State Feedback and Partial Feedback Linearization for output decoupling and underaction restrictions. The underactuated nature and the use of some part of the Feedback Linearization control induce undesirable residual dynamics. A second order model-free Sliding-Mode is used to guarantee robust regulation, while preserving zero dynamic stability. Representative simulations provide appreciation of the validity of the proposed approach.

References

1. Avila Vilchis, J.C.: *Modélisation et Commande d'Hélicoptère*, PhD dissertation, Laboratoire d'Automatique de Grenoble (LAG), Septembre 2001, Institut National Polytechnique de Grenoble (INPG), France
2. Bergerman, Marcel; Amidi, Omead; Miller, James Ryan; Vallidis, Nicholas & Dudek, Todd: *Cascaded Position and Heading Control of a Robotic Helicopter*, Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, Oct 29 - Nov 2, 2007.
3. Chen, Chi-Tsong: *Linear System Theory and Design*, Third Edition, 1999, ISBN:0-19-511777-8.
4. Dzul, A.; Lozano, R. & Castillo: *Adaptative Altitude Control for a Small Helicopter in Vertical Flying Stand*, Proc. of IEEE Conference on Decision and Control, Maui, Dec. 2003.
5. Fossen, Thor Inge: *Guidance and Control of Ocean Vehicles*, Chichester, John Wiley and Sons Ltd, University of Trondheim, Norway, 1994.
6. Gavrilets, V.; Mettler, B. & Feron, E.: *Nonlinear model for a small-size acrobatic helicopter*, in Proceedings of AIAA Guidance Navigation and Control Conference, Montreal, Quebec, Canada, August 2001.
7. Isidori, Alberto: *Nonlinear Control Systems*, Springer-Verlag, London 2003, 3nd ed. ISBN: 3540199160
8. Lin, Feng: *Optimal Control Design: An Optimal Control Approach*, John Wiley & Sons, Ltd, 2007, ISBN:978-0-470-03191-9.
9. Mahony, R.; Hamel, T. & Dzul, A.: *Hover Control via Lyapunov Control for an Autonomous Model Helicopter*, Proc. of IEEE Conference on Decision and Control, Phoenix, Dec. 1999.
10. Olguín-Díaz, Ernesto; Rubio, Luis; Mendoza, Alejandro; Gómez, Nestor & Benes, Bedrich: *Animacin Virtual en 3D de un helicptero de R/C a escala*, 3ra Conferencia Iberoamericana en Sistemas, Ciberntica e Informtica (CISCI 2004). Orlando, FL, July 2004.
11. V. Parra-Vega, S. Arimoto, Y.H. Liu, G. Hirzinger, and P. Akella: *Dynamic sliding PID control for tracking of robot manipulators: Theory and experiments*, IEEE Transaction on Robotics and Automation, Vol. 19, No. 6, pages 976 985. December 2003.
12. Shim, D.H.; Koo, T.J.; Hoffmann, F. & Sastry, S.: *Comprehensive Study of Control Design for an Autonomous Helicopter*, Proc. of IEEE Conference on Decision and Control, Tampa. 1998.
13. Spong, M. W.: *Underactuated Mechanical Systems*, in Control Problems in Robotics and Automation, B. Siciliano and K. P. Valavanis Eds., LNCIS, vol. 230, pp. 135–150, Springer Verlag, London, 1998. http://citeseer.ist.psu.edu/spong98underactuated.html
14. Vite-Medécigo, S. & Olguín-Díaz, Ernesto: *Robust Tracking and Disturbances Rejection with Robust LQR via BDU for a R/C Helicopter*, CIIIEE 2008 - ITA, November 2008, ISBN:978-607-95060-1-8

Interests-Sensitive Data Dissemination Protocol for Vehicular Ad-hoc Networks

Amal El-Nahas

The German University in Cairo, New Cairo city, Egypt
amal.elnahas@guc.edu.eg

Abstract. Vehicular ad-hoc network (VANET) is a potential player in an intelligent transportation system that would increase road safety as well as road confort. In VANETs, vehicles excahnge road-related information either through an established infrastructure, which is costly, or through their collaboration when in common transmission range which we adopt in this paper. Information dissemination is realized through broadcasting, thus an intelligent selection technique should be deployed to decrease the traffic load caused by unnecessary rebroadcasting. In this paper, we propose an interest-aware data dissemination protocol that periodically exploits the current neighbors ínterests to select the proper set of data to be broadcasted. The proposed approach is structure-less and imposes minimum overhead on the communication bandwidth. The protocol is evaluated through simulation experiments and rResults obtained demonstrate that this approach maximizes the number of relevant data reports received by the vehicles, especially if a certain data type is more popular than the others.

1 Introduction

Vehicular ad-hoc networks (VANETs) have emerged as a result of the increased number of vehicles capable of wirelessly interconnecting through their onboard radio communication devices, thus forming an ad hoc network on the fly. Moreover, DSRC standard (Dedicated Short Range Communication), developed by IEEE, provides vehicular ad hoc networks with large bandwidth. Seven non-overlapping channels, each of 27 Mbps bandwidth, can be used for data dissemination. Only one channel is dedicated for safety messages while the rest can be used for other road-related services.

Data dissemination is performed either through vehicle to infrastructure communication or vehicle to vehicle communication. While the former requires the existing of an infrastructure in form of road side units which imposes additional cost and delay, the latter is purely based on the ability of vehicles within common transmission range to communicate. Multi-hop transmission is needed in order for the data to reach farther vehicles.

The easiness and self configuring nature of VANETs enabled a broad range of information applications ranging from road safety to journey comfort to appear. Vehicles collect and exchange information, in form of data reports, for traffic intensity, services along the road, weather conditions, free parking places and others. Thus, each vehicle can be a report producer, a report receiver, or both at the same time. It has been shown

that VANETs have a highly dynamic topology due to its highly mobile nodes. Consequently, vehicular ad hoc networks tend to be often sparsely connected. Thus, data reports received by a vehicle are most likely to be stored for a while before being retransmitted when encountering a new neighbor. Limiting the number of stored reports and selecting the most important ones for transmission is a challenge that attracted research lately. Intelligent dissemination protocols should be adopted to decide which reports to store and rebroadcast later so as to efficiently share the wireless bandwidth as well as decrease the amount of unwanted messages received by the drivers.

Data dissemination has long been studied for mobile users with short range wireless communication forming a mobile ad-hoc network (MANET). Various protocols have been suggested in the literature, the simplest of all is one that relies on flooding. Each moving node broadcasts data to all its neighbors until either covering the whole network or reaching the maximum number of hops. This uncontrolled simple flooding approach leads to increasing the number of unnecessary data retransmission, causing what is known by the broadcast storm that results in inefficient bandwidth utilization and severe congestion, as observed in [4]. Consequently, a constrained flooding approach should be implemented. Different improvements over the basic flooding approach have been proposed in the literature that either control the time when to rebroadcast, or apply rules to decide whether to rebroadcast or not. A comprehensive survey can be found in [1,2,3]. Relying on the observation that travelers tend to have individual preferences in the type of content of data reports they would prefer to receive, we propose in this paper incorporating drivers ' interests in the selection of data reports to be broadcasted. Reports are assumed to belong to one of predefined service categories. Neighbors ínterest in each category is locally computed at each vehicle. Most certainly, exploring the continuously changing neighbors ínterests without imposing extra overhead is not trivial. However, our protocol uses the periodic transmission of the beacon messages generated by the medium access control protocol for interests advertisement. The proposed protocol is evaluated through simulation experiments and proved to maximize the number of relevant information received by the vehicles.

The rest of the paper is organized as follows. Section 2 presents the different dissemination approaches suggested before. Section 3 presents an overview of the proposed interests-sensitive dissemination algorithm. In section 4, the simulation experiments and the analysis of the obtained results are discussed. Finally, section 5 concludes the paper with proposals for future work.

2 Data Dissemination Approaches

In general, data dissemination approaches proposed for ad-hoc networks considered one or more of the three main resource constraints in MANETs, namely communication bandwidth, energy consumption and storage [5]. Controlled flooding were suggested to decrease the number of repetitive retransmissions, thus efficiently use the bandwidth as well as decrease the energy consumption on the mobile devices. Control is performed either by the sender or the receiver. Receiver-based approaches proved to perform well in MANETs. In [6], data is forwarded to all nodes within a specific area defined by the sender. Each receiving node decides whether to rebroadcast the data or not based

on its physical location with respect to the specified area. An improvement over this geographic-based approach were proposed in [7] where a content-based forwarding approach (CBF) is defined. In this approach, the next node for forwarding the message is selected by all neighboring nodes based on their actual position when receiving the message using a contention process. The farthest node that most likely would reach the destination is the best one considered for forwarding the data. Other variants exist in the literature. A comprehensive survey can be found in [1, 3].

Although the above techniques succeeded in reducing traffic load, they did not take into consideration the node ' s content requirements. A more selective approach should be used to intelligently select the set of data reports to be forwarded to prevent users from receiving unwanted messages. In [8], Wolfson et al. proposed a spatio-temporal selection approach that is based on the data novelty probability. The novelty probability of a data report reflects how new, and hence useful, this report is for the recipients based on its generation time and distance to generation location. As time or distance or both increases, report ages and eventually disappears. Although this approach proved to be efficient in terms of throughput and response time, it did not consider the individual users interests in the novelty probability. An autonomous gossiping approach for ad hoc networks were proposed in [9] where information is sent only to neighbors interested in receiving it. Each node advertises its profile that defines its interest. In addition, each data item maintains its own profile. Based on nodes and data item profiles, data items decide whether to replicate to a better node, migrate or do nothing.

However, it is worthy to note that VANETs have unique and challenging features that do not exist in MANETs. Examples of such features are the highly dynamic topology, highly mobile nodes, time critical responses, and insensitivity to energy consumption and computation power that are considered unlimited As a result, data dissemination protocols specifically designed for VANETs have been proposed. In [10], Tonguz et al. propose using the traffic density as well as the covered distance to decide whether to retransmit or not. In a dense area, only a subset of cars needs to rebroadcast. Moreover, as distance increase between the source of information and the node, the frequency of broadcasting is decreased. A similar approach, but taking time into consideration, was proposed in [11] where nodes receive data and store it for later retransmission. Only fresh data is rebroadcasted. Combining both distance and time in a relevance function, which extend the idea of Wolfson previously proposed for MANETs to VANETs, is introduced in [12, 13]. AutoCast in [14] uses a probabilistic flooding, that depends on neighborhood size. Individual interests in data disseminated were taken into consideration in some recent work. In [15], messages selected are based on their benefit to expected recipients. The benefit depends on the message context, vehicle context and information context. A different approach in [16] is based on a pull model where a node uses an utility-based approach to determine which data to pull upon meeting another node.

3 Interests-Sensitive Data Dissemination Algorithm

The interests-sensitive dissemination algorithm we propose is a structure-less algorithm that does not rely on any existing infrastructure. It is based on the same model as in [12,

13] to send and receive data reports, but augmented with an interest level component that collects neighbors ínterests using the MAC layer single hop beacon messages. We assume that data reports belong to one of predefined service categories based on their content. Categories may represent traffic data, parking service, weather information, and many others. Each vehicle has its own interest in each of those categories based on their content. Each vehicle has its own interest in each of those categories. The vehicle model and protocol description are presented below.

3.1 Vehicle Model

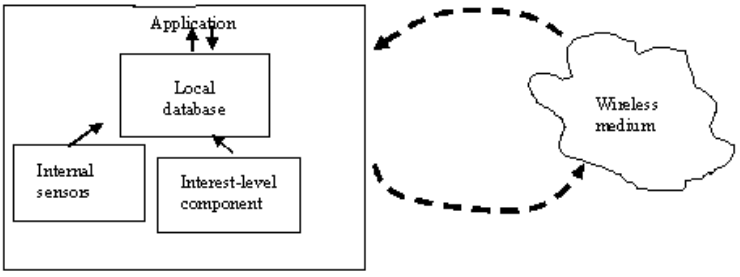


Fig. 1. Vehicle Model.

We consider a network formed by a number of moving vehicles modeled as in Figure 1. Each vehicle is equipped with a GPS receiver for localization, integrated sensors to collect road-related data, such as road surface condition, speed, light intensity or others, as well as a wireless radio communication interface to communicate with other vehicles. Each moving vehicle is modeled to have unlimited processing power but limited storage capacities. A local database is maintained at each vehicle that is limited in size to M reports, where M is a configurable parameter. The interest level component calculates and stores the interest level of current neighbors for each category. In the example shown in Table 1, 65 percent of the current neighbors are interested in receiving traffic related reports, 25 percent are interested in weather conditions, while 10 percent are interested in information about availability of parking places. This information is collected from the beacon messages periodically generated by each vehicle and saved in the node ś neighbor table.

Table 1. Interests Levels for 3 categories.

Category	Interest level
Traffic conditions	0.65
Weather conditions	0.25
Parking places	0.1

3.2 Interests-Sensitive Algorithm

As mentioned before, a vehicle can generate, send or receive data reports. A data report is composed of the following tuples:
carid, report-id (ID), generation time (t), generation location in x (x), generation location in y (y), category-id (CAT)

where category-id represents the service category the record belongs to. Report generation occurs upon detection of variations in the vehicle sensed data, like change in speed, road surface, or others. In this case, a report is generated and added to the vehicle local database either for immediate or later transmission. Sending data reports occurs upon either detecting a new vehicle in the neighborhood, or having a new data report generated that needs to be sent immediately. In this case, the vehicle checks its local database and selects the most relevant reports to be broadcasted. The relevance of the reports is based on the current neighbors ś interest level that is calculated for each category (i) by dividing the number of nodes interested in i, $(nodes_i)$, by the total number of neighboring nodes, as follows.

$$intLevel(i) = \frac{\sum nodes_i}{\sum neighbouringNodes} \tag{1}$$

The number of nodes interested in each category is calculated from the information saved in the neighbors table updated with the recipient of each beacon message, while the interest level is calculated upon transmitting a data report.

Lastly receiving data reports occurs when in range with any of the neighboring vehicles transmitting. The received set of reports is checked against the stored one and new reports are then added to the database. Figure 2 illustrates a simplified pseudo code for the proposed algorithm.

4 Simulation Methodology and Experimental Results

To further prove our concept, we simulated the behavior of the proposed protocol using Vsim, a VANET simulator created in the University of Ulm, Germany [17]. The simulator used combines both a road traffic simulation with a communication simulation as discussed below. In the following subsections, we present the simulation methodology, then the analysis of the results obtained.

4.1 Traffic Model

Traffic simulation is based on the traffic model of Nagel and Schreckenberg [18] where vehicles are generated randomly from the roads endpoints, heading to randomly chosen destinations. Their velocity and position are updated every 100 msec taking into consideration the rules for changing lanes and the behavior at intersections. In our experiment, a single bidirectional road model was used for testing. Vehicles are generated from both ends and move in opposite directions.

```
begin
    while (true) {
    If (change in sensed data > threshold) {
        compose report;
        add to localDB;
        }
    TransmitData();
    ReceiveData();
    }
end.
TransmitData() {
    if (new neighbor || timer expires) {
       for each category (i)
           calculate interest level;
       sort database descendingly;
       transmit top R records;
    }
ReceiveData()  {
    if (receive report from neighbor){
       for each report i in localDB {
          if (receivedreport == report.i)
              discard;
       }
       insert in localDB
    }
```

Fig. 2. Pseudo-code of the algorithm.

4.2 Communication Model

In this model, vehicles are communicating using 802.11 standard, where every 100 ms each vehicle broadcasts a beacon message to exchange its state with the surrounding neighborhood. A beacon message (HELLO message) is used by each node to build its own neighbor table. Each HELLO message is of length 105 bytes: 25 bytes for the header and 80 bytes for the data. The message header contains the car-ID, generation time, (x,y) coordinates of the vehicle and a list of its categories of interests. We limited our model to only 4 categories, as discussed below. The transmission range is set to 500m.

4.3 Experimental Results and Analysis

We consider a single road with vehicles generated from both ends in opposite directions. Data reports are generated by only 20 percent of the vehicles, which represent an injection rate of 0.2. Only four categories for data reports were defined in our simulation: traffic condition, road services, weather and no preferences. Each vehicle interest is selected randomly amongst those categories with different probabilities. We conducted two experiments, one with uniform distribution amongst different categories by setting the all probability values to 25 percent. In the second experiment, we simulated the

scenario where 50 percent of vehicles were interested in traffic conditions, 20 percent in weather conditions, 20 percent in available gas stations and 10 percent in available parking places, as in table 2. Those values are tuning parameters that can be adjusted.

Table 2. Interests Levels for 4 categories.

Category	Interest level
Traffic conditions	0.5
Weather conditions	0.2
Gas stations	0.2
Parking places	0.1

Data report has a fixed size of 100 bytes and the local database size is fixed to 200 reports. The simulation experimented were conducted for a total simulation time of 30 minutes that is divided into steps, each of 100 msec length. The performance measure chosen for evaluation is the percentage of relevant reports received by the vehicles. It is calculated as the percentage of the number of relevant reports out of the total number of received reports. In Figures 3 and 4, the average percentage value for vehicles belonging to the same category obtained by applying our approach is plotted against the basic approach were relevance is not considered.

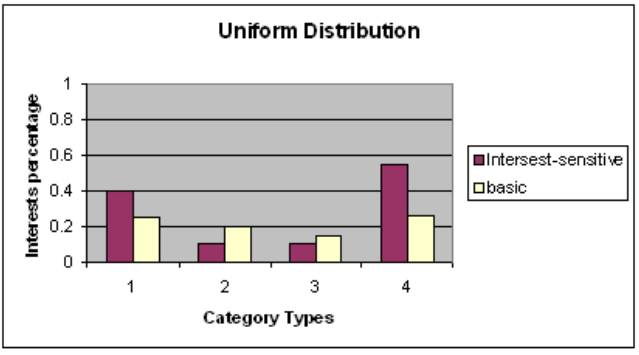


Fig. 3. Interest level per category, equiprobable interests.

Figure 3 represents the case where all categories have the same interest probability. It is clear that our approach has no benefits over the basic one as all categories are the same. As in Figure 4, when a certain category is of more interest to most of the vehicles, the improvement is clear for those vehicles. The percentage of relevant report received approaches 99 percent, while the rest of the vehicles experience decrease in their percentage. This is due to the selection process applied that selects only the top 10 relevant reports from the local database for transmission.

In order to enhance the performance of our approach, a better selection technique could be applied. The 10 reports selected for transmission should be selected from the 4

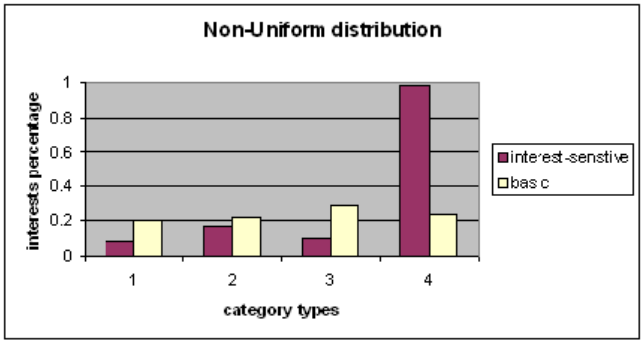


Fig. 4. Interest level per category, category 4 most popular.

different categories with respect to the percentage of their interest level at transmission time. In this case, and according the the values chosen in table 2, the ten selected data reports will consist of five reports belonging to category 1, two reports from category 2, two reports from category 3 and one report from category 4. Applying this technique is currently being investigated.

5 Conclusions

In this paper, we presented how we can substantially increase the percentage of relevant data reports received using a selective dissemination protocol that considers the vehicles índividual interests. With the application of our interest-sensitive protocol, up to 99 percent of the reports received were of interest to the users belonging to the most popular service category.Our experiments and results proved our concept. However more investigation needs to be conducted. Currently, we are testing the improved selection procedure to include non-popular categories as well. Furthermore, a city model is used for testing the effect of the city traffic on the overall performance.

Acknowledgements

Our thanks to Prof. Hans-Peter Grossmann and his group, especially Bernhard Wiegel from the University of Ulm, Germany, for their valuable support and stimulating discussions.

References

1. Williams, B., Camp. T.: Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks. ACM MOBIHOC (2002) 194–205
2. Krishnamurthy, P.: Information Dissemination and Information Assurance in Vehicular ad-hoc Networks: A Survey. Proc. of 3rd annual iConference (2008)
3. Hanbali, A., Ibrahim, M., Simon, V., and Endre: A Survey of Message Diffusion in Mobile Ad-hoc Networks. Proc. of International Conference on Performance, InterPerf (2008)

4. Ni, S., Tseng, Y., Chen, Y., Sheu, J.: The Broadcast Storm Problem in a Mobile Ad hoc Network. Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (1999) 152–162
5. Wolfson, O., Xu, B., Tanner, R.: Mobile peer-to-peer dissemination with resource constraints. 8th International conference on mobile data management, MDM (2007)
6. Ko, Y., Vaidya, N.: Flooding-based geocasting protocols for mobile ad hoc networks. Mobile Networks and Applications (2002) 471–480
7. Fussler, H., Widmer, J., Kasemann, M., Mauve, M., Hartenstein, H. :Contention-based forwarding for mobile ad hoc networks. Elsevierś ad hoc networks (2003) 351–369
8. Luo, Y., Wolfson, O., Xu, B.,: A spatio-temporal approach to selective data dissemination in mobile peer-to-peer networks. 3rd International conference on wireless and mobile communication, ICWMC (2007)
9. Datta, A.: Autonomous gossiping: a self-organizing epidemic algorithm for selective information dissemination in wireless mobile ad-hoc networks. International conference on semantics of a networked world , ICSNW (2004)
10. Tonguz, O., Wisitpongphan, N., Bai, F., Mudalige, P., and Sadekar, V. : Broadcasting in VANET. Proceedings of the 2007 Workshop on Mobile Networking for Vehicular Environments. (2007) 7–12
11. Dornbush, S., Joshi, A.: StreetSmart Traffic: Discovering and disseminating automobile congestion using vanets. 65th IEEE Vehicular Technology Conference (2007) 11–15
12. Xu, B., Ouksel, A., Wolfson, O.: Opportunistic resource exchange in inter-vehicle ad-hoc networks. MDM (2004)
13. Zhong, T., Wolfson, O.: Disseminating real-time traffic information in vehicular ad hoc networks. IVS (2008)
14. Wegener, A., Hellbrck, H., Fischer, S., Schmidt, C., Fekete, S.: AutoCast: An adaptive data dissemination protocol for traffic information systems. In VTC (2007) 1947–1951
15. Eichler, S., Schroth, C., Kosh. T., Strassberger, M., : Strategies for context adaptive message dissemination in vehicular ad hoc networks. V2VCOM (2006)
16. Boldrini, C., Conti, M., Passarella, A.:Context and resource awareness in opportunistic data dissemination. AOC (2008)
17. Gunter, Y., Grossmann, H., Khalifa, W., Salah, W., Karam, O.: Simulator for inter-vehicle communication based on traffic modeling. International Conference on Intelligent vehicles (2004)
18. Nagel, K., Schreckenberg, M. : A cellular automation model for freeway traffic. Jornal of Physique I, (1992) 2221–2229
19. Wischhof, L., Rohling, H. : On utility fair broadcast in vehicular ad-hoc networks. 2nd international workshop on intelligent transportation, Germany. (2005)
20. Adler, C., Eigner, R., Schroth, C., Strassberger, M. :Context-adaptive information dissemination in vanets: maximizing the global benefit. (2006)

Dynamic Routing using Real-time ITS Information

Ali R. Güner, Alper Murat and Ratna Babu Chinnam

Department of Industrial and Manufacturing Engineering, Wayne State University
4815 Fourth St. Detroit, MI 48202, U.S.A.
{arguner, amurat, r_chinnam}@wayne.edu

Abstract. On-time delivery is a key performance measure for dispatching and routing of freight vehicles in just-in-time (JIT) manufacturing environments. Growing travel time delays and variability, attributable to increasing congestion in transportation networks, are negatively impacting the efficiency of JIT logistics operations. Recurrent congestion is one of the primary reasons for delivery delay and variability. In this study, we propose a stochastic dynamic programming formulation for dynamic routing of vehicles in non-stationary stochastic networks subject to recurrent congestion. Results are very promising when the algorithms are tested in a simulated network of Southeast-Michigan freeways using historical Intelligent Transportation Systems (ITS) data.

1 Introduction

Supply chains that rely on just-in-time (JIT) production and distribution require timely and reliable freight pick-ups and deliveries from the freight carriers in all stages of the supply chain. However, road transportation networks are experiencing ever growing travel time delays, which greatly hinders all travel and certainly the freight delivery performance. Travel time delays are mostly attributable to the so called 'recurrent' congestion that, for example, develops due to high volume of traffic seen during peak commuting hours. The standard approach to deal with congestion is to build additional 'buffer time' into the trip (i.e., starting the trip earlier so as to end the trip on time). Intelligent Transportation Systems (ITS) are providing real-time traffic data (e.g., lane speeds and volumes) in many urban areas. In-vehicle communication technologies, such as satellite navigation systems, are also enabling drivers' access to this information en-route. In this paper, we precisely consider JIT pickup/delivery service, and propose a dynamic vehicle routing model that exploits real-time ITS information to avoid recurrent congestion.

Our problem setting is the non-stationary stochastic shortest path problem with recurrent congestion. We propose a dynamic vehicle routing model based on a Markov decision process (MDP) formulation. Stochastic dynamic programming is employed to derive the routing 'policy', as static 'paths' are provably suboptimal for this problem [1]. The MDP 'states' cover vehicle location, time of day, and network congestion state(s). Recurrent network congestion states and their transitions are estimated from the ITS historical data. The proposed framework employs Gaussian mixture model based clustering to identify the number of states and their transition rates, by time of day, for each arc of the traffic network. To prevent exponential

growth of the state space, we also recommend limiting the network monitoring to a reasonable vicinity of the vehicle.

The rest of the paper is organized as follows. Survey of relevant literature is given in section 2. Section 3 establishes modeling recurrent congestion and dynamic vehicle routing for the problem. Section 4 presents experimental settings and discusses the results. Finally, section 5 offers some concluding remarks.

2 Literature Survey

Shortest path problems with stochastic and time-dependent arc costs (STD-SP) are first studied by Hall [1]. Hall showed that the optimal solution has to be an 'adaptive decision policy' (ADP) rather than a single path. Hall [1] employed dynamic programming (DP) approach to derive the optimal policy. Later, Fu [2] discussed real-time vehicle routing based on the estimation of immediate arc travel times and proposed a label-correcting algorithm as a treatment to the recursive relations in DP. Waller and Ziliaskopoulos [3] suggested polynomial algorithms to find optimal policies for stochastic shortest path problems with one-step arc and limited temporal dependencies. For identifying paths with the least expected travel (LET) time, Miller-Hooks and Mahmassani [4] proposed a modified label-correcting algorithm. Miller-Hooks and Mahmassani [5] extends [4] by proposing algorithms that find the expected lower bound of LET paths and exact solutions by using hyperpaths.

All of the studies on STD-SP assume deterministic temporal dependence of arc costs, with the exception of [3] and [6]. Polychronopoulos and Tsitsiklis [7] is the first study to consider stochastic temporal dependence of arc costs and to suggest using online information en route. They defined environment state of nodes that is learned only when the vehicle arrives at the source node. They considered the state changes according to a Markovian process and employed a DP procedure to determine the optimal policy. Kim et al. [8] studied a similar problem as in [7] except that the information of all arcs are available real-time. They proposed a DP formulation where the state space includes states of all arcs, time, and the current node. They stated that the state space of the proposed formulation becomes quite large making the problem intractable. They reported substantial cost savings from a computational study based on the Southeast-Michigan's road network. To address the intractable state-space issue, Kim et al. [9] proposed state space reduction methods. A limitation of Kim et al.[8], is the modeling and partitioning of travel speeds for the determination of arc congestion states. They assume that the joint distribution of velocities from any two consecutive periods follows a single unimodal Gaussian distribution, which cannot adequately represent arc travel velocities for arcs that routinely experience multiple congestion states. Moreover, they also employ a fixed velocity threshold *(50 mph)* for all arcs and for all times in partitioning the Gaussian distribution for estimation of state-transition probabilities (i.e., transitions between congested and uncongested states). As a result, the value of real-time information is compromised rendering the loss of performance of the dynamic routing policy. Our proposed approach addresses all of these limitations.

3 Modeling

3.1 Recurrent Congestion Modeling

Let the graph $G=(N,A)$ denote the road network where N is the set of nodes (intersections) and $A \subseteq N \times N$ is the set of directed arcs between nodes. For every node pair, $n', n \in N$, there exists an arc $a \equiv (n,n') \in A$, if and only if, there is a road that permits traffic flow from node n to n'. Given an origin, n_0 -destination, n_d node (OD) pair, the trip planner's problem is to decide which arc to choose at each decision node such that the expected total trip travel time is minimized. We formulate this problem as a finite horizon Markov decision process (MDP), where the travel time on each arc follows a non-stationary stochastic process.

An arc is labeled as observed if its real-time traffic data (e.g., velocity) is available through the traffic information system. An observed arc can be in $r+1 \in \mathrm{Z}^+$ different states that represent arc's traffic congestion level at a time. We begin with discussing how to determine an arc's congestion state given the real-time velocity information and defer the discussion on estimation of the congestion state parameters to Section 4. Let $c_a^{i-1}(t)$ and $c_a^i(t)$ for *i=1,2,...,r+1* denote the cut-off velocities used to determine the state of arc a given the velocity at time t on arc a, $v_a(t)$. We further define $s_a(t)$ as the state of arc a at time t, i.e. $s_a(t) = \{\text{Congested at level } i\} = \{i\}$ and can be determined as: $s_a(t) = \{i, \text{if } c_a^{i-1}(t) \le v_a(t) < c_a^i(t)\}$. For instance, if there are two congestion levels (e.g., $r+1=2$), then the states will be i.e., $s_a(t) = \{\text{Uncongested}\} = \{0\}$ and $s_a(t) = \{\text{Congested}\} = \{1\}$ and the travel time is normally distributed at each state.

We assume the state of an arc evolves according to a non-stationary Markov chain. In a network with all arcs observed, $S(t)$ denotes the traffic congestion state vector for the entire network, i.e., $S(t) = \{s_1(t), s_2(t), ..., s_{|A|}(t)\}$ at time t. For presentation clarity, we will suppress (t) in the notation whenever time reference is obvious from the expression. Let the state realization of $S(t)$ be denoted by $s(t)$.

It is assumed that arc states are independent from each other and have the single-stage Markovian property. In order to estimate the state transitions for each arc, two consecutive periods' velocities are modeled jointly. Accordingly, the time-dependent single-period state transition probability from state $s_a(t)=i$ to state $s_a(t+1)=j$ is denoted with $P\{s_a(t+1)=j \mid s_a(t)=i\} = \alpha_a^{ij}(t)$. The transition probability for arc a, $\alpha_a^{ij}(t)$, is estimated from the joint velocity distribution as follows:

$$\alpha_a^{ij}(t) = \frac{\left| c_a^{i-1}(t) \le V_a(t) < c_a^i(t) \cap c_a^{j-1}(t+1) < V_a(t+1) < c_a^j(t+1) \right|}{\left| c_a^{i-1}(t) \le V_a(t) < c_a^i(t) \right|}$$

Let $T_a(t,t+1)$ denote the matrix of state transition probabilities from time t to time $t+1$, then we have $T_a(t,t+1)=\left[\alpha_a^{ij}(t)\right]_{ij}$. Note that the single-stage Markovian assumption is not restrictive for our approach as we could extend our methods to the multi-stage case by expanding the state space [10]. Let network be in state $S(t)$ at time t and we want to find the probability of the network state $S(t+\delta)$, where δ is a positive integer number. Given the independence assumption of arcs' congestion states, this can be formulated as follows:

$$P\left(S(t+\delta)\mid S(t)\right)=\prod_{a=1}^{|A|}P\left(s_a(t+\delta)\mid s_a(t)\right).$$

Then the congestion state transition probability matrix for each arc in δ periods can be found by the Kolmogorov's equation:

$$T_a(t,t+\delta)=\left[\alpha_a^{ij}(t)\right]_{ij}\times\left[\alpha_a^{ij}(t+1)\right]_{ij}\times\ldots\times\left[\alpha_a^{ij}(t+\delta)\right]_{ij}.$$

With the normal distribution assumption of velocities, the time to travel on an arc can be modeled as a non-stationary normal distribution. We further assume that the arc's travel time depends on the congestion state of the arc at the time of departure (equivalent to the arrival time whenever there is no waiting). It can be determined according to the corresponding normal distribution:

$$\delta(t,a,s_a)\sim N\left(\mu(t,a,s_a),\sigma^2(t,a,s_a)\right),$$

where $\delta(t,a,s_a)$ is the travel time; $\mu(t,a,s_a)$ and $\sigma(t,a,s_a)$ are the mean and the standard deviation of the travel time on arc a at time t with congestion state $s_a(t)$.

3.2 Dynamic Routing Model with Recurrent Congestion

We assume that the objective of our dynamic routing model is to minimize the expected travel time based on real-time information where the trip originates at node n_0 and concludes at node n_d. Let's assume that there is a feasible path between (n_0,n_d) where a path $p=(n_0,..,n_k,..,n_{K-1})$ is defined as sequence of nodes such that $a_k\equiv(n_k,n_{k+1})\in A$, $k=0,..,K-1$ and K is the number of nodes on the path. We define set $a_k\equiv(n_k,n_{k+1})\in A$ as the *current arc* set of node n_k, and denoted with $CrAS(n_k)$. That is, $CrAS(n_k)\equiv\{a_k : a_k\equiv(n_k,n_{k+1})\in A\}$ is the set of arcs emanating from node n_k. Each node on a path is a decision stage (or epoch) at which a routing decision (which node to select next) is to be made. Let $n_k\in N$ be the location of k^{th} decision stage, t_k is the time at k^{th} decision stage where $t_k\in\{1,...,T\}$, $T>t_{K-1}$. Note that we are discretizing the planning horizon.

While optimal dynamic routing policy requires real-time consideration and projection of the traffic states of the complete network, this approach makes the state space prohibitively large. In fact, there is little value in projecting the congestion

states well ahead of the current location. This is because the projected information is not different than the long run average steady state probabilities of the arc congestion states. Hence, an efficient but practical approach would tradeoff the degree of look ahead (e.g., number of arcs to monitor) with the resulting projection accuracy and routing performance. This has been very well illustrated in Kim et al. [9]. Thus we limit our look ahead to finite number of arcs that can vary by the vehicle location on the network. The selection of the arcs to monitor would depend on factors such as arc lengths, value of real-time information, and arcs' congestion state transition characteristics. For ease of presentation and without loss of generality, we choose to monitor only two arcs ahead of the vehicle location and model the rest of the arcs' congestion states through their steady state probabilities. Accordingly, we define the following two sets for all arcs in the network. $ScAS(a_k)$, the successor arc set of arc a_k, $ScAS(a_k) \equiv \{a_{k+1} : a_{k+1} \equiv (n_{k+1}, n_{k+2}) \in A\}$, i.e., the set of outgoing arcs from the destination node (n_{k+1}) of arc a_k. $PScAS(a_k)$, the post-successor arc set of arc a_k, $PScAS(a_k) \equiv \{a_{k+2} : a_{k+2} \equiv (n_{k+2}, n_{k+3}) \in A\}$ i.e., the set of outgoing arcs from the destination node (n_{k+2}) of arc a_{k+1}.

Since the total trip travel time is an additive function of the individual arc travel times on the path plus a penalty function measuring earliness/tardiness of arrival time to the destination node, the dynamic route selection problem can be modeled as a dynamic programming model. The state, $\left(n_k, t_k, s_{a_{k+1} \cup a_{k+2}, k}\right)$, of the system at k th decision stage is denoted by Ω_k. This state vector is composed of the state of the vehicle and network and thus characterized by the current node (n_k), the current node arrival time (t_k), and $s_{a_{k+1} \cup a_{k+2}, k}$ the congestion state of arcs $a_{k+1} \cup a_{k+2}$ where $\{a_{k+1} : a_{k+1} \in ScAS(a_k)\}$ and $\{a_{k+2} : a_{k+2} \in PScAS(a_k)\}$ at k th decision stage. The action space for the state Ω_k is the set of current arcs of node n_k, $CrAS(n_k)$.

At every decision stage, the trip planner evaluates the alternative arcs from $CrAS(n_k)$ based on the remaining expected travel time. The expected travel time at a given node with the selection of an outgoing arc is the expected arc travel time on the arc chosen and the expected travel time of the next node. Let $\pi = \{\pi_0, \pi_1, ..., \pi_{K-1}\}$ be the policy of the trip and is composed of policies for each of the *K-1* decision stages. For a given state $\Omega_k = \left(n_k, t_k, s_{a_{k+1} \cup a_{k+2}, k}\right)$, the policy $\pi_k(\Omega_k)$ is a deterministic Markov policy which chooses the outgoing arc from node n_k, i.e., $\pi_k(\Omega_k) = a \in CrAS(n_k)$. Therefore the expected travel cost for a given policy vector π is as follows:

$$F^{\pi}(\Omega_0) = \underset{\delta_k}{E}\left\{\sum_{k=0}^{K-2} g(\Omega_k, \pi_k(\Omega_k), \delta_k) + \overline{g}(\Omega_{K-1})\right\},$$

where $\Omega_0 = (n_0, t_0, S_0)$ is the starting state of the system. δ_k is the random travel

time at decision stage k, i.e., $\delta_k \equiv \delta\left(t_k, \pi_k\left(\Omega_k\right), s_a\left(t_k\right)\right)$. $g(\Omega_k, \pi_k\left(\Omega_k\right), \delta_k)$ is cost of travel on arc $\pi_k\left(\Omega_k\right) = a \in CrAS\left(n_k\right)$ at stage k, i.e., if travel cost is a function (ϕ) of the travel time, then $g(\Omega_k, \pi_k\left(\Omega_k\right), \delta_k) \equiv \phi\left(\delta_k\right)$ and $\overline{g}\left(\Omega_{K-1}\right)$ is terminal cost of earliness/tardiness of arrival time to the destination node under state Ω_{K-1}. Then the minimum expected travel time can be found by minimizing $F\left(\Omega_0\right)$ over the policy vector π as follows:

$$F^*\left(\Omega_0\right) = \min_{\pi=\{\pi_0,\pi_1,\ldots,\pi_{K-1}\}} F\left(\Omega_0\right).$$

The corresponding optimal policy is then $\pi^* = \arg\min_{\pi=\{\pi_0,\pi_1,\ldots,\pi_{K-1}\}} F\left(\Omega_0\right)$. Hence, the Bellman's cost-to-go equation for the dynamic programming model can be expressed as follows [10]:

$$F^*\left(\Omega_k\right) = \min_{\pi_k} E_{\delta_k}\left\{g(\Omega_k, \pi_k\left(\Omega_k\right), \delta_k) + F^*\left(\Omega_{k+1}\right)\right\}.$$

For a given policy $\pi_k\left(\Omega_k\right)$, we can re-express the cost-to-go function by writing the expectation in the following explicit form:

$$F\left(\Omega_k \mid a_k\right) = \sum_{\delta_k} P\left(\delta_k \mid \Omega_k, a_k\right)\Big[g\left(\Omega_k, a_k, \delta_k\right) + \sum_{s_{a_{k+1},k+1}} P\left(s_{a_{k+1},k+1}\left(t_{k+1}\right) \mid s_{a_{k+1},k}\left(t_k\right)\right) \sum_{s_{a_{k+2},k+1}} P\left(s_{a_{k+2},k+1}\left(t_{k+1}\right)\right) F\left(\Omega_{k+1}\right)\Big]$$

where $P\left(\delta_k \mid \Omega_k, a_k\right)$ is the probability of travelling arc a_k in δ_k periods. $P\left(s_{a_{k+2},k+1}\left(t_{k+1}\right)\right)$ is the long run probability of arc $a_{k+2} : a_{k+2} \in PScAS\left(a_k\right)$ being in state $s_{a_{k+2},k+1}$ in stage $k+1$. This probability can be calculated from the historical frequency of a state for a given arc and time.

We use backward dynamic programming algorithm to solve for $F_k^*\left(\Omega_k\right)$, $k = K-1, K-2, .., 0$. In the backward induction, we initialize the final decision epoch such that, $\Omega_{K-1} = \left(n_{K-1}, t_{K-1}, s_{K-1}\right)$, n_{K-1} is destination node, and $F_{K-1}\left(\Omega_{K-1}\right) = 0$ if $t_{K-1} \le T$. Accordingly, a penalty cost is accrued whenever there is delivery tardiness, e.g., $t_{K-1} > T$ .Note that $s_{K-1} = \varnothing$ since destination node current and successor arcs doesn't have value of information.

4 Experimental Studies

In this section we first introduce two road networks for demonstrating the performance of the proposed algorithms along with a description of their general traffic conditions. Then describe the process of how to model recurrent congestion. Finally, we report savings from employing the proposed model.

We test our procedure on a road network from South-East Michigan (Fig. 1). The sample network covers major freeways and highways in and around the Detroit metropolitan area. The network has 30 nodes and a total of 98 arcs with 43 observed arcs and 55 unobserved arcs. Real-time traffic data for the observed arcs is collected by Michigan ITS Center for 23 weekdays from January 21, 2008 to February 20, 2008 for the full 24 hours of each day at a resolution of an observation every minute. A small part of our full network, labeled *sub-network* (Fig. 1b), with 5 nodes and 6 observed arcs is used here to better illustrate the methods and results.

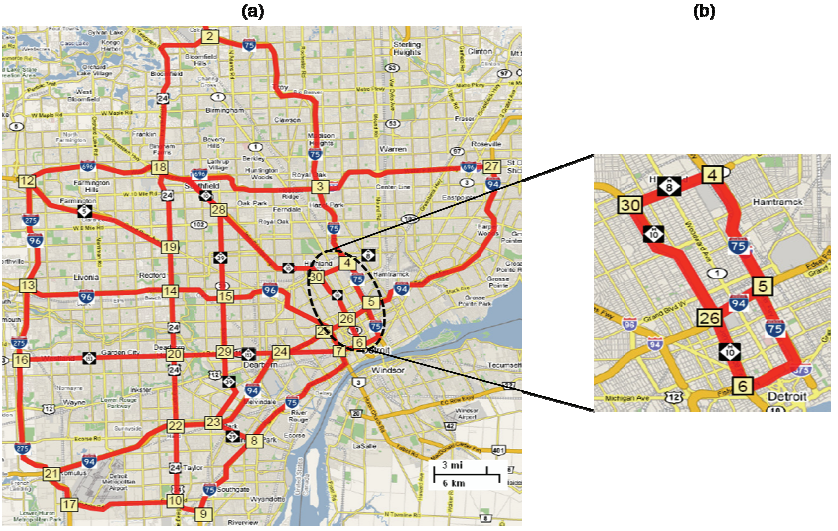


Fig. 1. (a) South-East Michigan road network considered for experimental study. (b) Sub-network from South-East Wayne County.

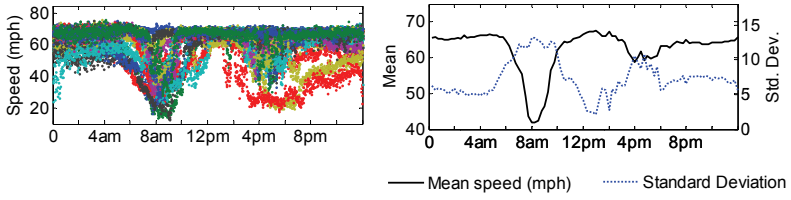


Fig. 2. For arc 4-to-5 (a) raw traffic speeds for 23 weekdays (b) mean (mph) and standard deviations of speeds by the time of day with 15 minute time interval resolution.

We present the speed data for arc 4-to-5 for the given days in Fig. 2.a as an example. The mean and standard deviations of speed for the arc 4-to-5 is also illustrated (Fig. 2.b). It can be seen clearly that the traffic speeds follow a stochastic non-stationary distribution that vary with the time of the day.

Given the traffic speed data, we employed the Gaussian Mixture Model (GMM) clustering technique to determine the number of recurrent-congestion states for each arc by time of day. In particular, we employed the greedy learning GMM clustering method of Verbeek [11] for its computational efficiency and performance. The

parameters of the traffic state joint Gaussian distributions (i.e., $\boldsymbol{\mu}^{i}_{t,t+1}; \boldsymbol{\Sigma}^{i}_{t,t+1}$) along with the computed cut-off speeds (if GMM yields more than one state) are employed to calculate travel time distribution parameters and the transition matrix elements as explained in section 3. In the event that two states are identified by GMM, α_t denotes the probability of state transition from congested state to congested state whereas β_t denotes the probability of state transition from uncongested state to uncongested state. Fig. 3a plots these transition rates for the arc 4-to-5 with a 15 minute time interval resolution. The mean travel time of arc 4-to-5 for congested and uncongested traffic states are given in Fig. 3b.

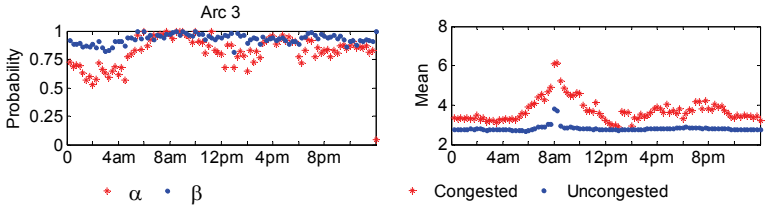


Fig. 3. For arc 4-to-5 (a) recurrent congestion state-transition probabilities where α: congested to congested transition; β: uncongested to uncongested transition probability; (b) mean travel time for congested and uncongested traffic states.

In the experiments based on the sub-network, node 4 is considered as the origin node and node 6 as the destination node of the trip. As stated earlier, we consider *node 4* as the origin node and *node 6* as the destination node of the trip. Three different path options exist (*path 1: 4-5-6; path 2: 4-5-26-6;* and *path 3: 4-30-26-6*). Given the historical traffic data, *path1: 4-5-6* dominates other paths most of the time of a day under all network states. Hence we identify *path 1* as the *baseline path* and show the savings (averaged over 10,000 runs) from using the proposed dynamic routing algorithm with regard to baseline path. Fig. 4a plots the corresponding percentage savings from employing the dynamic vehicle routing policy over the baseline path for each network traffic state combination and Fig. 4b shows the average savings (averaged across all network traffic states, treating them equally likely). It is clear that savings are higher and rather significant during peak traffic times and lower when there is not much congestion, as can be expected.

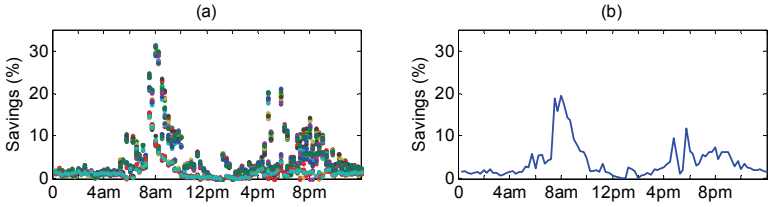


Fig. 4. (a) Savings for each of 32 network state combinations and (b) average savings for all state combinations during different times of the day.

Besides the sub-network (Fig. 1b), we have also randomly selected 4 other origin

and destination (OD) pairs (OD pair **1**: 2-21, **2**: 12-25, **3**: 19-27, and **4**: 23-13) to investigate the potential savings from using real-time traffic information under a dynamic routing policy. Once again, we identify the *baseline path* for each OD pair (as explained for the case of routing on the *sub-network*) and show percentage savings in mean travel times (over 10,000 runs) over the baseline paths from using the dynamic routing policy. The savings, Fig. 5, are consistent with results from the *sub-network*, further validating the sub-network results.

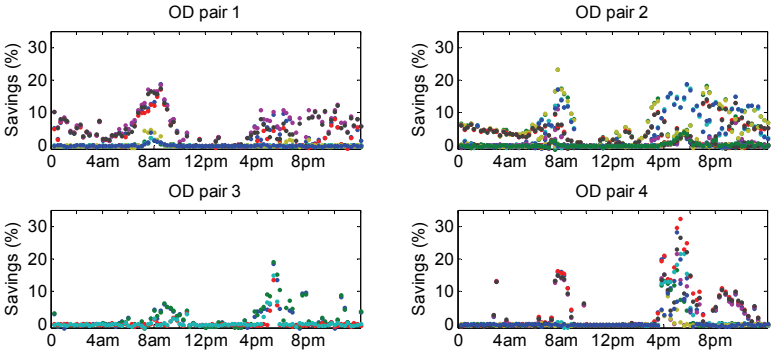


Fig. 5. Savings of dynamic policy over baseline path during the day for all starting states of given OD pairs of full network (with 15 minute time interval resolution).

5 Conclusions

The paper proposes practical dynamic routing models that can effectively exploit real-time traffic information from ITS regarding recurrent congestion in transportation networks. With the aid of this information and technologies, our models can help drivers avoid or mitigate trip delays by dynamically routing the vehicle from an origin to a destination in road networks. We model the problem as a non-stationary stochastic shortest path problem under recurrent congestion. We propose effective data driven methods for accurate modeling and estimation of recurrent congestion states and their state transitions.

ITS data from South-East Michigan road network, collected in collaboration with Michigan ITS Center, is used to illustrate the performance of the proposed models. Experiments show that as the uncertainty (standard deviation) in the travel time information increases, the dynamic routing policy that takes real-time traffic information into account becomes increasingly superior to static path planning methods. The savings however depend on the network states as well as the time of day. The savings are higher during peak times and lower when traffic tends to be static (especially at nights).

Acknowledgements

This work was supported by funds from the US Department of Transportation and the Michigan Department of Transportation through the MIOH University Transportation Center.

References

1. Hall, R.: The fastest path through a network with random time-dependent travel time. Transportation Science 20 (1986) 182-188
2. Fu, L.: An adaptive routing algorithm for in vehicle route guidance systems with real-time information. Transportation Research Part B 35B (2001) 749-765
3. Waller, S.T., Ziliaskopoulos, A.K.: On the online shortest path problem with limited arc cost dependencies. Networks 40 (2002) 216-227
4. Miller-Hooks, E.D., Mahmassani, H.S.: Least possible time paths in stochastic, time-varying networks. Computers & Operations Research 12 (1998) 1107-1125
5. Miller-Hooks, E.D., Mahmassani, H.S.: Least expected time paths in stochastic, time-varying transportation networks. Transportation Science 34 (2000) 198-215
6. Gao, S., Chabini, I.: Optimal routing policy problems in stochastic time-dependent networks. Transportation Research Part B 40B (2006) 93-122
7. Psaraftis, H.N., Tsitsiklis, J.N.: Dynamic shortest paths in acyclic networks with Markovian arc costs. Operations Research 41 (1993) 91-101
8. Kim, S., Lewis, M.E., White III, C.C.: Optimal vehicle routing with real-time traffic information. IEEE Transactions on Intelligent Transportation Systems 6 (2005) 178-188
9. Kim, S., Lewis, M.E., White III, C.C.: State space reduction for non-stationary stochastic shortest path problems with real-time traffic congestion information. IEEE Transactions on Intelligent Transportation Systems 6 (2005) 273-284
10. Bertsekas, D.P.: Dynamic Programming and Optimal Control. Athena Scientific (2001)
11. Verbeek, J.J., Vlassis, N., Kröse, B.: Efficient Greedy Learning of Gaussian Mixture Models. Neural Computation 5 (2003) 469-485

A Model of an Interregional Logistic System for the Statement and Solution of Decision Problems at the Operational Level

Chiara Bersani, Davide Giglio, Riccardo Minciardi, Michela Robba
Roberto Sacile, Simona Sacone and Silvia Siri

Department of Communications, Computer and Systems Science
University of Genova, Via Opera Pia 13, 16145 Genova, Italy
{chiara.bersani,davide.giglio,riccardo.minciardi
michela.robba,roberto.sacile,simona.sacone,silvia.siri}@unige.it

Abstract. A regional/multi-regional logistic traffic network is considered in this paper with the aim of optimizing the flows of goods which pass through the network in order to reach their final destinations. The logistic network takes into account both road and rail transportation, and it is modelled as a directed graph whose arcs represent a road or a rail link and whose nodes are not only connection points but can represent a place where some service activities (such as the change in transportation mode) are carried out. In the paper, the model of the logistic network and, in particular, the equations which formalize the dynamics of links and nodes, are described in detail. In addition, with reference to decision problems at operational level, some considerations about the degrees of freedom (decision variables) in the model, the kind and the role of decision makers, and the class of performance indicators are also outlined in the paper.

1 Introduction

Modelling, planning, and control of logistic systems are research streams that, in the last years, have received a significant attention by the research community due to their economic impact. An improvement of the performance of the overall logistic chain and an effective integration of the different actors of a logistic system are fundamental goals in the management of modern production/distribution systems. As a matter of fact, these systems have to be designed and planned to fulfil such relevant objectives as those related to the on-time delivery of products to final users, to the minimization of transportation costs and of costs referred to the use of infrastructures, etc.

In this context, off-line planning methodologies play a key role and a wide bibliography can be found on such subjects. Some interesting review works [1–4] define the hierarchical decisional structure to be used when dealing with systems devoted to freight intermodal transportation and, then, with logistic systems. This structure is composed of three levels: long term (or strategic) planning, medium term (or tactical) planning, and short term (or operational) planning. At the strategic level, planning problems are mainly relevant to demand forecasting, logistic nodes location [5, 6] and to the design of transportation operations between nodes [7, 8]. The tactical level consists in

the aggregate planning of operations in logistic nodes [9] and of distribution operations (Service Network Design problems [10]). Many decision problems are typically defined at the operational level and such problems require the adoption of several models and decision techniques; typical decision problems at this level are the assignment of transportation operations to transportation means [11] and the static and dynamic routing of vehicles on the transportation network or on the logistic chain [12, 13]. The model and the problems considered in this paper refer to this latter decision level.

In this paper, the model of a logistic traffic network at regional/multi-regional level is presented, being the final objective of the current research activity the statement and solution of decision problems for the management of a logistic system at operational level, such as the optimal routing of goods which pass through the logistic network in order to reach their final destinations. The proposed model is a discrete-time model and the time horizon to be considered can range from some hours to some days. The model mainly consists of a directed graph whose arcs represent a road or a rail link and whose nodes are not only connection points but can represent a place where some service activities (such as the change in transportation mode) are carried out. The model is based on some characteristics which have been introduced in [14] with reference to the macroscopic modelling of transportation networks. In particular, each link and some nodes of the logistic network are discrete-time dynamic systems whose input and output variables are represented by flows that are respectively received from and transmitted to the neighbouring links/nodes, and the basic dynamic equation is represented by the vehicle conservation equation introduced in [15, 16]. In addition, with reference to decision problems at operational level, some considerations about the degrees of freedom (decision variables) in the model, the kind and the role of decision makers, and the class of performance indicators are also outlined in the conclusions of the paper.

2 The Model of the Logistic Network

The model of the logistic network mainly consists of the transportation offer (i.e., the physical network where vehicles can move), the transportation demand (i.e., the requirements of moving goods over this network) and the equations that represent the dynamics of this system, both referred to nodes and links. The model is a discrete-time model; in this connection, let t and Δ be the generic time instant and the length of one interval, respectively, with $t = 0, \ldots, T$ being $T\Delta$ the time horizon. Note that, for the quantities considered in the model which are not referred to a time instant but to a time interval, with t we refer to the time interval $[t, t + 1)$.

2.1 The Transportation Offer

The offer of transportation services is represented by means of a directed graph $\mathcal{D} = (\mathcal{V}, \mathcal{A})$ where $\mathcal{V}$ is the set of nodes and $\mathcal{A}$ is the set of links. We will refer to each node as $i \in \mathcal{V}$ and to each link as the pair of nodes it connects, i.e. $(i, j) \in \mathcal{A}$. For each node $i \in \mathcal{V}$ the sets $\mathcal{P}(i)$ and $\mathcal{S}(i)$ gather the predecessor and successor nodes, respectively.

The graph $\mathcal{D}$ represents an intermodal network involving two transportation modes corresponding to *road* and *rail*. Let us denote with $\mathcal{A}^{\mathrm{R}}$ and $\mathcal{A}^{\mathrm{T}}$ the set of arcs on road

and on rail, respectively. It is $\mathcal{A}^{\mathrm{R}} \cap \mathcal{A}^{\mathrm{T}} = \emptyset$ since an arc corresponds univocally to a given transportation mode. Moreover, it is obvious that $\mathcal{A}^{\mathrm{R}} \cup \mathcal{A}^{\mathrm{T}} = \mathcal{A}$.

The nodes of the network are primarily divided into *connection nodes* and *service nodes*. The former are simply interconnections among different links and do not have their own dynamics, whereas the latter represent a place where some service activities are carried out (such as intermodal terminals where cargo is handled and there is a change in the transportation mode) and then are modelled as discrete-time dynamic systems. Both connection and service nodes can be either *regular nodes* or *border nodes*. Border nodes represent the access and exit points of the network. In this connection let $\mathcal{V}^{\mathrm{RC}}$, $\mathcal{V}^{\mathrm{BC}}$, $\mathcal{V}^{\mathrm{RS}}$, and $\mathcal{V}^{\mathrm{BS}}$ be, respectively, the set of regular connection, border connection, regular service, and border service nodes. These sets are disjoint ($\mathcal{V}^{\mathrm{RC}} \cap \mathcal{V}^{\mathrm{BC}} \cap \mathcal{V}^{\mathrm{RS}} \cap \mathcal{V}^{\mathrm{BS}} = \emptyset$) and their union correspond to the whole set of nodes ($\mathcal{V}^{\mathrm{RC}} \cup \mathcal{V}^{\mathrm{BC}} \cup \mathcal{V}^{\mathrm{RS}} \cup \mathcal{V}^{\mathrm{BS}} = \mathcal{V}$).

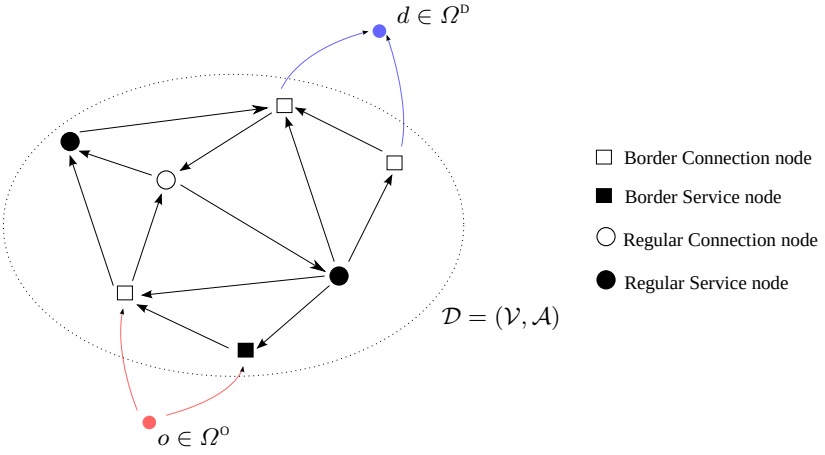


Fig. 1. A sketch of the logistic network.

2.2 The Transportation Demand

In the considered model, we suppose that the real origins and destinations of the demand are outside the transportation network $\mathcal{D}$. However, all goods must pass through the proposed regional/multi-regional logistic traffic network in order to reach their final destinations. At this purpose, let Ω^{O} and Ω^{D} represent, respectively, the set of origins and the set of destinations for the whole demand (see Fig. 1). Note that there can be some geographic areas that are both the origin and the destination of logistic flows, then in general $\Omega^{\mathrm{O}} \cap \Omega^{\mathrm{D}} \neq \emptyset$. Goods coming from a certain origin may enter the network through one or more "compatible" border nodes; in the same way goods can reach their destination by exiting the network from one or more "compatible" border nodes. Then, let $\mathcal{V}_o^{\mathrm{IN}} \subseteq \mathcal{V}^{\mathrm{BC}} \cup \mathcal{V}^{\mathrm{BS}}$ (resp., $\mathcal{V}_d^{\mathrm{OUT}} \subseteq \mathcal{V}^{\mathrm{BC}} \cup \mathcal{V}^{\mathrm{BS}}$) be the set of border nodes associated with origin $o \in \Omega^{\mathrm{O}}$ (resp., destination $d \in \Omega^{\mathrm{D}}$). Moreover, for each destination $d \in \Omega^{\mathrm{D}}$ and for each node $\mu \in \mathcal{V}_d^{\mathrm{OUT}}$, we denote with $\tau_{\mu,d}(t)$ the time necessary to reach d from μ if the logistic units are in μ at time t.

The transportation demand is defined for each different network user, i.e., road carrier, shipper and so on, that needs to transport some logistic units from a certain origin to a certain destination. Each network user is denoted with $n = 1, \ldots, N$ and it has a set of Γ_n transportation requests to satisfy. The l-th request of user n, $n = 1, \ldots, N$, $l = 1, \ldots, \Gamma_n$, is characterized by: *origin* $o_{n,l} \in \Omega^{\mathrm{O}}$, *destination* $d_{n,l} \in \Omega^{\mathrm{D}}$, *number of logistic units* $\delta_{n,l}$, *due date* $dd_{n,l}$, *release time* $rt_{n,l}$, i.e., the time instant in which the logistic units are available to enter the network. In addition, let $st_{n,l}$ be the time instant in which the logistic units actually enter the network; moreover, $\lambda_{n,l}^{\nu,\mu}$, $\nu \in \mathcal{V}_{o_{n,l}}^{\mathrm{IN}}$, $\mu \in \mathcal{V}_{d_{n,l}}^{\mathrm{OUT}}$ represents the percentage of $\delta_{n,l}$ that enter the network in ν and exit from μ. Note that these last two terms are decision variables whose values depend on the choices taken by the network user.

Finally, in order to associate the request l of network user n with the considered time horizon, let the function of time $\delta_{n,l}(t)$ be defined as follows:

$$\delta_{n,l}(t) = \begin{cases} \delta_{n,l} & \text{if } t = st_{n,l} \\ 0 & \text{otherwise} \end{cases} \qquad n = 1, \ldots, N \; l = 1, \ldots, \Gamma_n \; t = 0, \ldots, T \tag{1}$$

3 The Dynamics of the Logistic Network

Links and nodes are considered as discrete-time dynamic systems whose state is represented by the number of logistic units which are in the link or node at a certain time instant. Each state variable is updated according to a state equation (*conservation equation*) which takes into account the number of logistic units entering and exiting the link or node in the time interval between two subsequent time instants. Moreover, in order to separately consider all requests of all network users and all exiting nodes, an approach similar to the one proposed in [14], which considers destination-oriented variables (composition and splitting rates), is adopted.

3.1 Links

The dynamics of links involves road links only, since trains transporting a finite number of logistic units over a rail link are not explicitly modelled. As it will be clear in the following, the dynamics of trains is implicitly considered in the dynamics of service nodes. Then, in the following, it is assumed $(i, j) \in \mathcal{A}^{\mathrm{R}}$.

Let us denote with $n_{i,j}^{n,l,\mu}(t)$, $n = 1, \ldots, N$, $l = 1, \ldots, \Gamma_n$, $\mu \in \mathcal{V}_{d_{n,l}}^{\mathrm{OUT}}$, $t = 0, \ldots, T$, the number of logistic units, belonging to the l-th transportation request of network user n, which are in link (i, j), at time t, and have to reach border node μ. In the following, the triple (n, l, μ) will be referred to as a whole. The state equation is then given by:

$$n_{i,j}^{n,l,\mu}(t+1) = n_{i,j}^{n,l,\mu}(t) + q_{i,j}^{n,l,\mu}(t) - Q_{i,j}^{n,l,\mu}(t) \tag{2}$$

where $q_{i,j}^{n,l,\mu}(t)$ and $Q_{i,j}^{n,l,\mu}(t)$ are, respectively, the number of logistic units of (n, l, μ) which enter and exit (i, j) in the time interval $[t, t+1)$.

$Q_{i,j}^{n,l,\mu}(t)$ is given by:

$$Q_{i,j}^{n,l,\mu}(t) = \gamma_{i,j}^{n,l,\mu}(t) \cdot Q_{i,j}(t) \tag{3}$$

being the overall number of logistic units exiting from (i,j), namely $Q_{i,j}(t)$, obtained from

$$Q_{i,j}(t) = v_{i,j}(t) \cdot \rho_{i,j}(t) \cdot \Delta \tag{4}$$

where $v_{i,j}(t)$ and $\rho_{i,j}(t)$ indicate the mean speed and the density on link (i,j) in the time interval $[t, t+1)$. If we suppose that the density is uniformly distributed along (i,j) and constant in $[t, t+1)$, we can define the density as:

$$\rho_{i,j}(t) = \frac{n_{i,j}(t) + \overline{m}_{i,j}(t)}{L_{i,j}} \tag{5}$$

where $L_{i,j}$ is the length of (i,j) and $\overline{m}_{i,j}(t)$ represents the number of other vehicles (such as cars or other logistic vehicles which are not matter of decision in the considered system) present in (i,j) at time t. The value of $\overline{m}_{i,j}(t)$ is supposed to be known, at least as an average value, and then it is an input to the problem. However, note that $\overline{m}_{i,j}(t)$ must be taken into account because it affects the traffic behaviour and, then, the evolution of the state variable.

Moreover, the mean speed on the link is defined as $v_{i,j}(t) = f[\rho_{i,j}(t), (i,j), t]$, i.e., it is a function of the density on the link (as well as function of the link itself and of the time instant). This relation is generally known as the *steady state speed-density characteristic* [17].

The *link composition rate* $\gamma_{i,j}^{n,l,\mu}(t)$ specifies the fraction of logistic units, which are actually in link (i,j), belonging to (n,l,μ), with respect of the overall number of logistics units in (i,j). It is computed as

$$\gamma_{i,j}^{n,l,\mu}(t) = \frac{n_{i,j}^{n,l,\mu}(t)}{n_{i,j}(t)} = \frac{n_{i,j}^{n,l,\mu}(t)}{\sum_{n=1}^{N} \sum_{l=1}^{\Gamma_n} \sum_{\mu \in \mathcal{V}_{d_{n,l}}^{\text{OUT}}} n_{i,j}^{n,l,\mu}(t)} \tag{6}$$

The equation providing $q_{i,j}^{n,l,\mu}(t)$ depends on the kind of node i. If node i is a regular connection node, then

$$q_{i,j}^{n,l,\mu}(t) = \sum_{h \in \mathcal{P}(i)} \beta_{h,i,j}^{n,l,\mu}(t) \cdot Q_{h,i}(t) \qquad i \in \mathcal{V}^{\text{RC}} \tag{7}$$

where $\beta_{h,i,j}^{n,l,\mu}(t)$ is the *link splitting rate* from link (h,i) to link (i,j), in the time interval $[t, t+1)$, with reference to (n,l,μ). The link splitting rates are given by

$$\beta_{h,i,j}^{n,l,\mu}(t) = \gamma_{i,j}^{n,l,\mu}(t) \cdot \alpha_{h,i,j}^{n,l,\mu}(t) \tag{8}$$

where $\alpha_{h,i,j}^{n,l,\mu}(t)$ are *route choice parameters*. If node i is a border connection node and represents one of the access points for the logistic units belonging to (n,l,μ) (that is, $i \in \mathcal{V}_{o_{n,l}}^{\text{IN}}$), then

$$q_{i,j}^{n,l,\mu}(t) = \beta_{i,j}^{n,l,\mu}(t) \cdot \lambda_{n,l}^{i,\mu} \cdot \delta_{n,l}(t) \qquad i \in \mathcal{V}_{o_{n,l}}^{\text{IN}} \subseteq \mathcal{V}^{\text{BC}} \tag{9}$$

where $\beta_{i,j}^{n,l,\mu}(t)$ is the *node splitting rate* from node i to link (i,j), in the time interval $[t,t+1)$, with reference to (n,l,μ). Finally, if node i is a service node, both regular and border, the dynamics of the node must be taken into account, thus

$$q_{i,j}^{n,l,\mu}(t) = \beta_{i,j}^{n,l,\mu}(t) \cdot \widetilde{Q}_i(t) \qquad i \in \mathcal{V}^{\text{RS}} \cup \mathcal{V}^{\text{BS}} \tag{10}$$

where $\widetilde{Q}_i(t)$ is the number of logistic units exiting the node i (see next subsection).

3.2 Nodes

The dynamics of nodes is related to the possibility of queuing logistic units inside the node and thus it involves service nodes only (both regular and border). Let us denote with $n_i^{n,l,\mu}(t)$, $n = 1,\ldots,N$, $l = 1,\ldots,\Gamma_n$, $\mu \in \mathcal{V}_{d_{n,l}}^{\text{OUT}}$, $t = 0,\ldots,T$, the number of logistic units, belonging to the l-th transportation request of network user n, which are in node i, at time t, and have to reach border node μ. As before, in the following, the triple (n,l,μ) will be referred to as a whole. The state equation is then given by:

$$n_i^{n,l,\mu}(t+1) = n_i^{n,l,\mu}(t) + q_i^{n,l,\mu}(t) - Q_i^{n,l,\mu}(t) \tag{11}$$

where $q_i^{n,l,\mu}(t)$ and $Q_i^{n,l,\mu}(t)$ are, respectively, the number of logistic units of (n,l,μ) which enter and exit i in the time interval $[t,t+1)$.

$Q_i^{n,l,\mu}(t)$ is given by

$$Q_i^{n,l,\mu}(t) = \widetilde{Q}_i^{n,l,\mu}(t) + \widehat{Q}_i^{n,l,\mu}(t) \tag{12}$$

where $\widetilde{Q}_i^{n,l,\mu}(t)$ (resp., $\widehat{Q}_i^{n,l,\mu}(t)$) represents the overall number of logistic units, belonging to (n,l,μ), exiting from node i and entering a road link (resp., rail link). $\widetilde{Q}_i^{n,l,\mu}(t)$ is provided by

$$\widetilde{Q}_i^{n,l,\mu}(t) = \widetilde{\gamma}_i^{n,l,\mu}(t) \cdot \widetilde{Q}_i(t) \tag{13}$$

where $\widetilde{\gamma}_i^{n,l,\mu}(t)$ is the *node-to-road composition rate*, and $\widetilde{Q}_i(t)$ is the overall number of logistic units exiting i and entering a road link; this last term is given by

$$\widetilde{Q}_i(t) = \min\left\{\widetilde{\sigma}_i(t) \cdot n_i(t), \widetilde{s}_i(t) \cdot \Delta\right\} \tag{14}$$

with

$$\widetilde{\sigma}_i(t) = \sum_{n=1}^{N} \sum_{l=1}^{\Gamma_n} \sum_{\mu \in \mathcal{V}_{d_{n,l}}^{\text{OUT}}} \widetilde{\sigma}_i^{n,l,\mu}(t) \tag{15}$$

being $\widetilde{\sigma}_i^{n,l,\mu}(t)$ the fraction of logistic units of (n,l,μ) which are in node i at time t and leave, in the subsequent time interval, namely $[t,t+1)$, the node towards a road link or leave the network, and

$$n_i(t) = \sum_{n=1}^{N} \sum_{l=1}^{\Gamma_n} \sum_{\mu \in \mathcal{V}_{d_{n,l}}^{\text{OUT}}} n_i^{n,l,\mu}(t) \tag{16}$$

Then, the node-to-road composition rate can be computed as

$$\widetilde{\gamma}_i^{n,l,\mu}(t) = \frac{\widetilde{\sigma}_i^{n,l,\mu}(t) \cdot n_i^{n,l,\mu}(t)}{\widetilde{\sigma}_i(t) \cdot n_i(t)} \tag{17}$$

In (14), $\widetilde{s}_i(t)$ represents the *node-to-road service rate* (expressed as number of logistic units per time unit) in the node i in the time interval $[t, t+1)$. Note that it is assumed that every logistic unit entering a service node in a given time interval cannot exit the node itself in the same time interval.

Before introducing the equation providing $\widehat{Q}_i^{n,l,\mu}(t)$, it is necessary to briefly describe the behaviour of logistic units on rail links. A rail link $(i,j) \in \mathcal{A}^{\mathrm{T}}$ is assumed to be served by one or more trains which transport logistic units from i to j. It is assumed that one train begins a transportation in i at each time instant and the number of logistic units that are transported by the train depends on the state of the node. However, such a number is upper-bounded by a value $C_{i,j}(t)$ which represents the capacity (maximum number of logistic units that can be transported) of the train leaving i towards j, at time instant t. Moreover, let $\Lambda_{i,j}$ be the travel time of a train travelling from i to j, expressed as number of time intervals; such a value is assumed fixed and a-priori known.

Because of the finite capacity of trains, some of the logistic units that concluded their service and that have to proceed with their travel in a rail link, may be not allowed to exit the node. Then, it is necessary to distinguish between the "potential" number of logistic units which leave from the node and the "actual" number. In (12), $\widehat{Q}_i^{n,l,\mu}(t)$, is the actual number. The potential number is provided by

$$\widehat{Q}_i^{\mathrm{POT}\,n,l,\mu}(t) = \widehat{\gamma}_i^{n,l,\mu}(t) \cdot \widehat{Q}_i^{\mathrm{POT}}(t) \tag{18}$$

where $\widehat{\gamma}_i^{n,l,\mu}(t)$ is the *node-to-rail composition rate*, and $\widehat{Q}_i^{\mathrm{POT}}(t)$ is the overall number of logistic units which potentially exit i and enter a rail link; this last term is given by

$$\widehat{Q}_i^{\mathrm{POT}}(t) = \min\left\{\widehat{\sigma}_i(t) \cdot n_i(t), \widehat{s}_i(t) \cdot \Delta\right\} \tag{19}$$

with

$$\widehat{\sigma}_i(t) = \sum_{n=1}^{N} \sum_{l=1}^{\Gamma_n} \sum_{\mu \in \mathcal{V}_{d_{n,l}}^{\mathrm{OUT}}} \widehat{\sigma}_i^{n,l,\mu}(t) \tag{20}$$

being $\widehat{\sigma}_i^{n,l,\mu}(t) = 1 - \widetilde{\sigma}_i^{n,l,\mu}(t)$, $\forall (n,l,\mu)$, the fraction of logistic units of (n,l,μ) which are in node i at time t and leave the node towards a rail link. Then, the node-to-rail composition rate can be computed as

$$\widehat{\gamma}_i^{n,l,\mu}(t) = \frac{\widehat{\sigma}_i^{n,l,\mu}(t) \cdot n_i^{n,l,\mu}(t)}{\widehat{\sigma}_i(t) \cdot n_i(t)} \tag{21}$$

In (19), $\widehat{s}_i(t)$ represents the *node-to-rail service rate* in the node i in the time interval $[t, t+1)$. The actual number of logistic units which leave from the node is then computed as

$$\widehat{Q}_i^{n,l,\mu}(t) = \sum_{\substack{j \in \mathcal{S}(i) \\ (i,j) \in \mathcal{A}^{\mathrm{T}}}} \widehat{\xi}_{i,j}^{n,l,\mu}(t) \cdot \widehat{Q}_i^{\mathrm{POT}\,n,l,\mu}(t) \tag{22}$$

where $\widehat{\xi}_{i,j}^{n,l,\mu}(t)$ represents the fraction of logistic units of (n,l,μ) which actually leave the node i towards rail link (i,j), with respect to the relative potential number. It is worth noting that the meaning of $\widehat{\xi}_{i,j}^{n,l,\mu}(t)$ is different from that of splitting rates introduced in the link dynamics. Moreover, such quantities must satisfy the following constraint

$$\sum_{n=1}^{N}\sum_{l=1}^{\Gamma_n}\sum_{\mu\in\mathcal{V}_{d_{n,l}}^{\text{OUT}}}\widehat{\xi}_{i,j}^{n,l,\mu}(t)\cdot\widehat{Q}_i^{\text{POT}\,n,l,\mu}(t)\leq C_{i,j}(t) \tag{23}$$

It is worth finally observing that, when $i=\mu$, all logistic units belonging to (n,l,μ) leave the network; in this case, it turns out $\widetilde{\sigma}_{\mu}^{n,l,\mu}(t)=1$, $\widehat{\sigma}_{\mu}^{n,l,\mu}(t)=0$, $t=0,\ldots,T$.

Coming back to (11), $q_i^{n,l,\mu}(t)$ is given by

$$q_i^{n,l,\mu}(t)=\begin{cases}\lambda_{n,l}^{i,\mu}\cdot\delta_{n,l}(t) & i\in\mathcal{V}_{o_{n,l}}^{\text{IN}}\subseteq\mathcal{V}^{\text{BS}}\\ \widetilde{q}_i^{n,l,\mu}(t)+\widehat{q}_i^{n,l,\mu}(t) & i\in\mathcal{V}^{\text{RS}}\cup\mathcal{V}^{\text{BS}},\, i\notin\mathcal{V}_{o_{n,l}}^{\text{IN}}\subseteq\mathcal{V}^{\text{BS}}\end{cases} \tag{24}$$

where, in case of service nodes that are not an access point for logistic units belonging to (n,l,μ) (bottom expression of (24)), $\widetilde{q}_i^{n,l,\mu}(t)$ (resp., $\widehat{q}_i^{n,l,\mu}(t)$) represents the overall number of logistic units, belonging to (n,l,μ), coming from a road link (resp., rail link) and entering node i. $\widetilde{q}_i^{n,l,\mu}(t)$ and $\widehat{q}_i^{n,l,\mu}(t)$ are provided by

$$\widetilde{q}_i^{n,l,\mu}(t)=\sum_{\substack{h\in\mathcal{P}(i)\\(h,i)\in\mathcal{A}^{\text{R}}}}Q_{h,i}^{n,l,\mu}(t) \tag{25}$$

$$\widehat{q}_i^{n,l,\mu}(t)=\sum_{\substack{h\in\mathcal{P}(i)\\(h,i)\in\mathcal{A}^{\text{T}}}}\widehat{\xi}_{h,i}^{n,l,\mu}(t-\Lambda_{i,j})\cdot\widehat{Q}_h^{\text{POT}\,n,l,\mu}(t-\Lambda_{i,j}) \tag{26}$$

4 Conclusions and Further Research Directions

In the previous section the model of an intermodal logistic network has been presented. The dynamic evolution of the elements (links and nodes) of this network has been represented by means of discrete-time state equations where the state variables indicate the number of logistic units present in a link or in a node. The main decisions to be taken concern the splitting of these logistic units over the alternative paths in the network (and consequently the choice of transportation mode) and the time instant in which they enter the network. Different approaches can be defined in order to determine these decisions and they depend on which decision makers are considered and, for each decision maker, the decision power, the available information and the performance indexes.

Three classes of decision makers can be considered in general. First of all, *network users* are decision makers that must move goods from given origins to given destinations, characterized by specific due dates. These network users work in a competitive environment, therefore each of them is characterized by a specific objective (i.e. minimizing costs and/or travel times in order to deliver goods within a given due date). Another class of decision makers is given by *infrastructure managers*, such as managers

of links (e.g. highways) or managers of nodes (e.g. terminal operators) or managers of trains. Each of them has, again, a specific objective (i.e. minimizing risk factors, maximizing profits, and so on) that can be in conflict with the objectives of other decision makers. A third class of decision makers is represented by the *local authorities* or *territory managers* devoted to manage the territory with social objectives (such as assuring security, minimizing traffic congestion, and so on). These three classes of decision makers are involved in a decision framework that is, in general, a hierarchic structure. The territory manager is at the top of this decision structure, it decides on the basis of its social objectives and it can act on the system in two ways, by advising the other decision makers about how to act or by imposing to them some policies (e.g. forbidding to cover a given link in a certain time period, imposing the number of specific cargo units that can move in a part of the network, and so on). The decisions taken by the territory manager affect the decisions of the network managers that, again, can be applied by advisory or coercive policies and, in their turn, affect the decisions of the network users. Therefore, the network users make their decisions by taking into account the social policies of the territory managers and the cost/incentive policies provided by the infrastructure managers.

The main decisions of the proposed system, i.e. the definition of the path followed by the logistic units, the transportation mode and the time instant in which they enter the network, are taken by network users and this can be obtained as the solution of a specific optimization problem. The considered objective function concerns the minimization of some cost terms concerning the network users (travel costs, also including highway or rail fares, deviations from due dates, and so on), possibly weighted in a different way for each network user. In the considered optimization problem, the constraints include the discrete-time state equations of nodes and links, as well as some other specific constraints. Note that the decisions taken by infrastructure managers and territory managers can affect the optimization problem both in the objective function and in the constraints. For instance, if the manager of a node/link applies different fares in different time slots, this is considered in the problem objective function. Otherwise, if the territory manager imposes a limit to the number of logistic units that can move in a certain area in a given time slot, this is considered in the problem by adding a constraint.

The proposed model is very general and can refer to different real applications, by adding specific constraints and/or decision variables. If a completely centralized system is considered, a single large optimization problem must be solved. Since such a problem generally has a nonlinear form, if real applications are considered, the problem dimensions are probably too large to be solved with nonlinear solvers. For this reason, it could be more reasonable to state different separate problems for each network user or for groups of network users, in order to obtain smaller instances of the problem. Anyway, in this case, it is necessary to model the interaction among the network users (either in a competitive or in a cooperative environment) such that an overall solution can be obtained by considering the single solutions that each of them has found by solving its specific optimization problem. The present research activity is devoted to the analysis of some real situations and the statement of ad-hoc optimization problems, in order to evaluate the effectiveness of different management policies in logistic networks.

References

1. C. J. Vidal and M. Goetschalckx, “Strategic production-distribution models: a critical review with emphasis on global supply chain models”, *European Journal of Operational Research*, vol. 98 (1997).
2. M. Goetschalckx, C. J. Vidal, and K. Dogan, “Modeling and design of global logistic systems: a review of integrated strategic and tactical models and design algorithms”, *European Journal of Operational Research*, vol. 143 (2002).
3. T. G. Crainic, “Long-haul freight transportation”, in R. W. Hall, editor, *Handbook of Transportation Science*, Kluwer Academic Publishers (2003).
4. C. F. Daganzo, *Logistics systems analysis*, 4th edition, Springer (2005).
5. H. Pirkul, V. Jayaraman, “A multicommodity, multiplan, capacitated facility location problem: formulation and efficient heuristic solution”, *Computers and Operations Research*, vol. 25, pp. 869-878 (1998).
6. M. S. Daskin and S. H. Owen, “Location models in transportation”, in R. W. Hall, editor, *Handbook of Transportation Science*, Kluwer Academic Publishers (2003).
7. T. G. Crainic and G. Laporte, “Planning models for freight transportation”, *European Journal of Operational Research*, vol. 98 (1997).
8. T. G. Crainic, “Network design in freight transportation”, *European Journal of Operational Research*, vol. 122 (2000).
9. T. G. Crainic and J. Roy, “O. R. tools for tactical freight transportation planning”, *European Journal of Operational Research*, vol. 33 (1988).
10. D. Kim, C. Barnhart, K. A. Ware, and G. Reinhardt, “Multimodal express package delivery: a service network design application”, *Transportation Science*, vol. 33 (1999).
11. W. B. Powell, “A comparative review of alternative algorithms for the dynamic vehicle allocation problem”, in B. L. Golden and A. A. Assad, editors, *Vehicle Routing: Methods and Studies*, North-Holland (1988).
12. W. B. Powell, P. Jaillet, and A. Odoni, “Stochastic and dynamic networks and routing”, in M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser, editors, *Handbooks in Operations Research and Management Science: Network Routing*, North-Holland (1995).
13. W. B. Powell, B. Bouzaiene-Ayari, and H. P. Simao, “Dynamic models for freight transportation”, in C. Barnhart and G. Laporte, editors, *Handbooks in Operations Research and Management Science: Transportation*, North-Holland (2006).
14. M. Papageorgiou, “Dynamic modeling, assignment an route guidance in traffic networks”, *Transportation Research B*, vol. 28-2, pp. 279-293 (1990).
15. M. H. Lighthill and G. B. Whitham, “On kinematics waves II: a theory of traffic flow on long, crowded roads”, *Proceedings of the Royal Society of London series A*, vol. 229, pp. 317-345 (1955).
16. P. I. Richards, “Shockwaves on the highway”, *Operations Research*, vol. 4, pp. 42-51 (1956).
17. D. R. Drew, *Traffic flow theory and control*, McGraw-Hill (1968).

Stability and Performance of Scheduling Policies in a Transportation Node

Mauro Boccadoro[1], Francesco Martinelli[2] and Paolo Valigi[1]

[1] DIEI, University of Perugia, Italy
{boccadoro,valigi}@diei.unipg.it
[2] DISP, University of Rome "Tor Vergata", Italy
martinelli@disp.uniroma2.it

Abstract. In this paper we consider the dynamic model of a logistic node of a transportation network and study dispatching feedback policies in terms of stability and optimality. A necessary and sufficient condition for the existence of a stable feedback policy is given and a policy is presented which would be optimal if the transportation resources were continuous.

1 Introduction

An intermodal logistic system can be modeled as a network comprising a set of nodes (hubs and terminals) connected by the links established by the transport operations, which, in general, take place under different modes. The management of logistic nodes in this network is a complex problem where several factors have to be taken into account, from the availability of carriers and their assignment to particular tasks (in terms of products to be shipped, destinations or routes), and fulfillment of various performance criteria such as timely delivery, minimization of transportation and inventory costs (possibly, both at the logistic nodes and at the destinations); see among others, [1–4].

Many instances of decisional problems for these systems are presented and solved in the literature; often transportation problems can be addressed in terms of linear programming problems, see e.g., [5–8], and developing ad-hoc techniques to obtain the solutions, such as dynamic programming with linear approximation of the (unknown) value function. It is worth mentioning that by the approach of [5, 6] the framework of the Logistic Queuing Networks is introduced.

A slightly different paradigm considers shipping policies for simplified models of a logistic network (or a part of it) and addresses the minimization of transportation and inventory costs, see for example [9, 10] and [15] where a stochastic setting is adopted. Still another example of management problems for a logistic node is represented by the optimization of space allocated for containers in ports (e.g., [13]); or the optimization of the operations of discharging containers from a ship, their location in the terminal yard and the upload of new containers [14]. In these last two cases, the performances considered can be also viewed in terms of the necessity to maintain low levels of stocked products in the logistic node (in this case, a port). The stability of the dynamics of the stock at a logistic node is therefore a relevant issue to be taken into account.

In order to study the stability (and the performance of stabilizing policies) from a dynamical point of view, in this paper a simplified model of a logistic node in a transportation network is considered and feedback policies based on the current state of the system are defined to control the node. The scenario is similar to those arising in other applicative domains (like in manufacturing, communications, computer systems or queuing networks in general), so the feedback policies considered in this paper have been inspired by various well established techniques developed in those domains. The stability of these policies will be investigated and a necessary and sufficient condition for the possibility of stabilizing the system will be determined. The sufficiency will be established in a constructive way by determining a class of policies which guarantee the stability of the system. A comparison among the performances of different stabilizing policies will be carried out through simulations, showing that a policy, inspired by a control known as optimal for the fluid version of the problem, will provide the best results among the policies considered in the paper.

2 Problem Formulation

Consider the discrete time model of a logistic node collecting Q different types of wares which have to be shipped to P different locations, and let $x_{ij}(k) \geq 0$ be the quantity of items of type $j = 1, \ldots, Q$, with destination $i = 1, \ldots, P$, stocked at the logistic node at time t_k, and collected in the buffer[3] B_{ij}. In this model a *destination* could be more in general considered as a *route* among different locations, established through some routing algorithm.

The time evolution of each x_{ij} is observed at various decisional time instants t_k, and as such characterized by a discrete time dynamics. Denoting $d_{ij}(k)$ the amount of goods of type j to be sent to destination i arriving in the node in the interval (t_k, t_{k+1}) and $u_{ij}(k)$ the amount of goods of type j shipped to destination i from the node in the same interval (t_k, t_{k+1}), we have:

$$x_{ij}(k+1) = x_{ij}(k) + d_{ij}(k) - u_{ij}(k) \tag{1}$$

In addition to this dynamics, we consider that of the vehicles executing the shipping task. Let $n_i(k)$ be the number of vehicles assigned to destination i in the interval (t_k, t_{k+1}); the total number $N(k)$ of vehicles present in the node at time t_k obeys to the following equation:

$$N(k+1) = N(k) + R(k) - \sum_{i=1}^{P} n_i(k) \tag{2}$$

where $R(k)$ is the number of vehicles arriving from outside in the interval (t_k, t_{k+1}). Notice that, according to the above dynamics, the total number of vehicles available for a shipping task at time t_k is given by

[3] These buffers could be considered as virtual, in the sense that in some cases we may have items which are physically stocked in different places according to their type (in such a way that the physical content of a buffer is given by $\sum_{i=1}^{P} x_{ij}$) as it happens for the stocked finished products in a factory.

$$N_a(k) := N(k) + R(k) \tag{3}$$

To model the inflow of vehicles $R(k)$, first consider the simple scenario where there is a fixed quantity N_c of vehicles which could serve the logistic node (as if for example, the logistic node is a shipper who owns a certain quantity N_c of trucks). A traveling time $T_i \in \mathbb{N}$ is associated to each route (i.e. destination) i; in particular T_i denotes the round-trip time, i.e. the interval after which a vehicle is again available at the node after completion of a shipping task to destination i. In this case $R(k) = \sum_{i=1}^{P} n_i(k - T_i)$ is the number of vehicles coming back from their expedition, and therefore (2) reads as follows:

$$N(k+1) = N(k) + \sum_{i=1}^{P} n_i(k - T_i) - \sum_{i=1}^{P} n_i(k); \qquad N(0) = N_c \tag{4}$$

Notice that the total sum of vehicles (those at the logistic node and those traveling) equals N_c at each time instant.

In other cases we can consider the logistic node and the shippers as separate entities, so that the total number of vehicles which are going to access the logistic node varies with time; in such situations N_c can be obtained through a suitable average of the expedition history in the node, and can be possibly perturbed when new vehicles are assigned to (or removed from) the node. A possible way to model this situation is by perturbing the signal $R(k)$, i.e., $R(k) = \sum_{i=1}^{P} n_i(k - T_i) + \Delta(k)$, where Δ is a disturbance signal characterized by certain statistical properties (e.g. zero mean). Another interesting extension would be to add to the round trip time some noise (possibly asymmetric, in the sense that positive perturbations, so that $\tilde{T}_i > T_i$, are more likely to occur than negative ones). As a first approach to the problem, in the following we consider the simplified model (4), i.e., assuming that the T_i's are deterministic quantities and N_c is fixed. In this case the number of available vehicles at time t_k is: $N_a(k) = N(k) + \sum_{i=1}^{P} n_i(k - T_i)$.

Let's now consider the interaction between the stock dynamics (1) and the vehicle dynamics (4). To this end, assume that each vehicle has identical volume capacity and that each item of type $j = 1, \ldots, Q$ has a relative volume with respect to vehicle capacity $v_j \leq 1$ (that is, a vehicle has unit capacity). Accordingly, we have the following constraint for any route i:q

$$\sum_{j=1}^{Q} v_j u_{ij}(k) \in [0, n_i(k)]. \tag{5}$$

Since $n_i(k)$ vehicles are used at time t_k for route i, it is reasonable that the above quantity is larger than $n_i(k) - 1$ (actually, by the policies that will be considered in this paper, vehicles travel completely loaded).

The objectives of this work will be essentially two. First, derive conditions on the *stability* of the system, that is conditions on the inflow process $d(\cdot)$, (relative) part volumes v_j, traveling times T_i and number of vehicles N_c such that there exists a policy of selection of $n_i(\cdot)$ and $u_{ij}(\cdot)$ which maintains limited all the buffers $x_{ij}(\cdot)$. Second, analyze the *performance* of some class of policies, trying to solve the optimization problem

consisting in the selection of the $n_i(k)$ and of the $u_{ij}(k)$ to minimize:

$$J = \sum_{k=1}^{K} g[x(k)]\gamma^k \tag{6}$$

where $\gamma \in (0, 1]$ is a discount factor and K a planning horizon, possibly infinite. The function $g(\cdot)$ penalizes waiting freights in the node, e.g., for a linear $g(x)$,

$$g(x) = \sum_{i=1}^{P}\sum_{j=1}^{Q} c_{ij}x_{ij} \tag{7}$$

We now make a fluid approximation for the variables involved in (1), considering x_{ij}, d_{ij} and u_{ij} as continuous quantities. Accordingly, the information about the volume of each type $j = 1, \ldots, Q$ is now carried by the continuous variables (now the relative volumes v_j have no sense per se, hence they will be dropped in the following) and each cost c_{ij}, assuming a fixed i, now has the meaning of holding cost of part $j = 1, \ldots, Q$ per unit volume[4]. Notice that also the variables N, N_c, n_i will represent volumes (multiples of the unit volume).

We will deal with the two problems above by restricting the control policies to those which make vehicles travel completely full (this is possible under the fluid approximation of the materials): this should represent, as remarked below, a correct choice under heavy traffic conditions. Notice, also, that transportation costs have not been included in the cost index: this depends on the fact that (i) transportation costs are considered constant in time; (ii) we restrict the analysis to policies which make all vehicles travel completely full. The assumptions above imply that the transportation cost is a fixed component that does not influence the optimization problem. The choice of considering the vehicles fully loaded is reasonable under heavy traffic conditions (where allowing the possibility of sending partially full vehicles may even compromise the stability), but may become significantly sub-optimal in the case of reduced inflow rates, large holding costs c_{ij} and small traveling costs.

3 Stability

As an introduction, consider a one part-type system ($Q = 1$) with constant inflow processes d_i and equal transportation times $T_i = T, \forall i$. The equations are then:

$$x_i(k+1) = x_i(k) + d_i - u_i(k),\ i = 1, \ldots, P \tag{8}$$

$$u_i(k) \in [0, n_i(k)] \tag{9}$$

$$\sum_{i=1}^{P} n_i(k) \leq N_a(k) \tag{10}$$

$$N(k+1) = N(k) + \sum_{i=1}^{P} n_i(k-T) - \sum_{i=1}^{P} n_i(k) \tag{11}$$

[4] Formally, as if the system were described by new variables $x'_{ij} = x_{ij}v_j$ (and similarly for d_{ij} and u_{ij}) and $c'_{ij} = c_{ij}/v_j$; dropping the "prime" and remaining with the same notation.

Based on the Little's law, the necessary and sufficient condition of stability for this system should be:

$$\sum_{i=1}^{P} d_i \leq \frac{N_c}{T} \tag{12}$$

In fact, N_c/T is actually the effective number of vehicles available at each time unit, and hence also the volume of goods the node may handle in each unit of time. This must be equal to the volume arriving from outside, i.e. $\sum_{i=1}^{P} d_i$.

The stronger condition that there exists a static vehicle allocation such that:

$$T\, d_i \leq n_i \tag{13}$$

for all i, which implies condition (12), actually is not necessary (but clearly sufficient, since if it holds, allows to apply a policy where vehicles are divided once for ever among the tasks and each task is fulfilled, with no interaction among them), as shown in the following simple example.

Example. Consider a system with $d_1 = d_2 = .5$, $T = 1$, $N_c = 1$. Clearly it is not possible to distribute vehicles once for ever (in fact for any static selection of n_i, condition (13) does not hold). However (12) holds and, in fact, the periodic allocation $n_1(\cdot) = \{1, 0, 1, \ldots\}$ and $n_2(k) = 1 - n_1(k)$, maintains the buffers bounded. □

Let us now return to the general case, but considering at first a constant inflow process. Condition (12) should be substituted by:

$$\sum_{i=1}^{P} \sum_{j=1}^{Q} d_{ij} T_i \leq N_c \tag{14}$$

which will be shown to be necessary and sufficient for the stability of the node. In this case, in fact, the quantity $d_{ij}T_i$ plays the role of a *work* inflow in the system per unit of time (in the sense that for each item of type j to be sent to i, the system must allocate a working capacity of T_i, where the total working capacity is N_c). In the case of time varying inflow rates (but with the inflow rate oscillating in a bounded interval), the same condition should hold with average inflow rates $\bar{d}_{ij}$.

Remark 1. Actually, while (14) is necessary for stability, the proof reported below only holds if the inequality in (14) is strict. We believe however that also the equality ensures the stability. Notice, in any case, that a strict inequality should be considered in practical settings to guarantee a certain degree of robustness of the stability property.

The previous discussion can be formalized in the following theorem.

Theorem 1. *Condition (14) is necessary and sufficient (if taken with strict inequality) to maintain all the buffers in the node bounded at all times.*

Proof. *Necessity.* The necessity of (14) can be shown by relaxing the integer constraint on the $n_i(k)$. If the vehicle resource is not discrete, it is possible to maintain all the buffers bounded only if there exists a static assignment of the vehicles (notice in fact that in our model the inflow process is constant) which balances the freight inflow into

the system for all the routes i. The freight inflow into the system of parts to be sent on the route i is given by $D_i := \sum_{j=1}^{Q} d_{ij}$. If n_i vehicles are assigned to this route, since each transport requires T_i time units, n_i vehicles are available only every T_i time units. The amount of wares accumulated in such a period is given by $D_i\,T_i$. So it must be $D_i\,T_i \leq n_i$. Summing over i, we get the condition (14). Since this condition is necessary for the relaxed problem, it is necessary also for the original problem.

Sufficiency. The proof of sufficiency is constructive: we exhibit a class of policies which, if (14) holds with strict inequality, ensures that all the buffers remain bounded. The proof is very similar to the proof of Theorem 1 in [11]. The class of policies ensuring stability is like the CAF policies in [11] where, however, the buffer x_{ij} is processed not until it is cleared (level zero) but until its level becomes lower than N_c. That is: all the vehicles are assigned to a single route by filling them with the products of a certain buffer B_{ij} (selected according to the CAF rule (15) reported below) only if this buffer has sufficient stock to use all vehicles, and the buffer is changed when this is no more possible. If no buffer can fill all the vehicles, the system remains idle until this becomes possible. Let τ_n denote the time a buffer has been finished to be processed. At each time τ_n the next buffer will be the one (denoted with a $*$) satisfying:

$$x^*(\tau_n) \geq \epsilon \sum_{i,j} x_{ij}(\tau_n) \tag{15}$$

for some $\epsilon > 0$ (e.g. the policy which selects the buffer with the largest content will belong to this class, satisfying (15) with any $\epsilon \in (0, 1/P)$, see [11]). Let $\bar{T}_i := T_i/N_c$. Performing a derivation similar to the one reported in [11], it is possible to show that:

$$\tau_{n+1} - \tau_n \leq \frac{\bar{T}^* x^*(\tau_n)}{1-\rho^*} + \frac{N_c}{d^*} \tag{16}$$

where the $*$ denotes the quantities corresponding to the buffer selected at time τ_n and $\rho^* := \bar{T}^* d^*$. The terms in (16) have been obtained as follows: the first term $\frac{\bar{T}^* x^*(\tau_n)}{1-\rho^*}$ corresponds to the time to bring the buffer x^* from its initial level $x^*(\tau_n)$ to a value below N_c and is derived from [11] setting the setup time δ to 0 and considering that we only need to reach a value below N_c and not 0; the second term $\frac{N_c}{d^*}$ takes into account that when a buffer is selected, perhaps its content is less than N_c. We define, as in [11]:

$$w(k) = \sum_{i,j} \bar{T}_i x_{ij}(k)$$

Then we have:

$$w(\tau_{n+1}) = \sum_{i,j} \bar{T}_i x_{ij}(\tau_{n+1}) =$$

$$= \sum_{i,j \neq *} \bar{T}_i \left[x_{ij}(\tau_n) + d_{ij}(\tau_{n+1} - \tau_n)\right] + \bar{T}^* x^*(\tau_{n+1})$$

$$= w(\tau_n) + \sum_{i,j \neq *} \bar{T}_i d_{ij}(\tau_{n+1} - \tau_n) + \bar{T}^* [x^*(\tau_{n+1}) - x^*(\tau_n)]$$

$$\leq w(\tau_n) + \sum_{i,j \neq *} \bar{T}_i d_{ij} (\tau_{n+1} - \tau_n) + \bar{T}^* N_c - \bar{T}^* x^*(\tau_n)$$

where the last inequality is implied by the fact that $x^*(\tau_{n+1}) \leq N_c$ (we stop processing x^* at time τ_{n+1}, when its content is below N_c). Exploiting (16),

$$w(\tau_{n+1}) \leq w(\tau_n) + \sum_{i,j \neq *} \bar{T}_i d_{ij} \left(\frac{\bar{T}^* x^*(\tau_n)}{1 - \rho^*} + \frac{N_c}{d^*} \right) + \bar{T}^* N_c - \bar{T}^* x^*(\tau_n)$$

Now, introducing the notation $\rho := \sum_{i,j} \bar{T}_i d_{ij}$, we have that $\sum_{i,j \neq *} \bar{T}_i d_{ij} = \rho - \rho^*$. Introducing this in the equation above and simplifying, we get:

$$w(\tau_{n+1}) \leq w(\tau_n) - \bar{T}^* x^*(\tau_n) \frac{1 - \rho}{1 - \rho^*} + \frac{\rho}{d^*} N_c$$

Using (15), the previous becomes:

$$w(\tau_{n+1}) \leq w(\tau_n) - \bar{T}^* \epsilon \sum_{i,j} x_{ij}(\tau_n) \frac{1 - \rho}{1 - \rho^*} + \frac{\rho}{d^*} N_c$$

$$\leq w(\tau_n) - \frac{\bar{T}^*}{\bar{T}_M} \epsilon \sum_{i,j} \bar{T}_i x_{ij}(\tau_n) \frac{1 - \rho}{1 - \rho^*} + \frac{\rho}{d^*} N_c$$

where $\bar{T}_M = \max_i \bar{T}_i$. So,

$$w(\tau_{n+1}) \leq w(\tau_n) \left[1 - \epsilon \frac{\bar{T}^*}{\bar{T}_M} \frac{1 - \rho}{1 - \rho^*} \right] + \frac{\rho}{d^*} N_c$$

Notice that condition (14) under strict inequality can be written as $\rho < 1$, which is exactly the condition considered in [11]. The proof can be continued exactly as in [11] where, however, for us $\alpha_{ij} = \bar{T}_i \frac{1-\rho}{1-\rho_{ij}}$ (the same as in [11]) and $\beta_{ij} = \frac{\rho}{d_{ij}} N_c$. So, as in [11], it is possible to obtain:

$$sup_n w(\tau_n) \leq \frac{\bar{T}_M}{\epsilon} \max_{ij} \frac{\beta_{ij}}{\alpha_{ij}}$$

hence

$$w(t_k) \leq \frac{\bar{T}_M}{\epsilon} \max_{ij} \frac{\beta_{ij}}{\alpha_{ij}} + \rho \frac{N_c}{d_m}$$

where $d_m = \min_{ij} d_{ij}$. This allows to obtain that

$$\sum_{ij} x_{ij}(t_k) \leq \frac{1}{\bar{T}_m} w(t_k) \leq \frac{\bar{T}_M}{\bar{T}_m \epsilon} \max_{ij} \frac{\beta_{ij}}{\alpha_{ij}} + \rho \frac{N_c}{\rho_m}$$

is bounded for all t_k (where $\bar{T}_m := \min_i \bar{T}_i$). □

4 Optimization

Consider for now $Q = 1$ (one part type system). Now, under condition (14), if everything is approximated through continuous variables, the optimal policy is myopic [12], that is, it is the $c\mu$ rule if dealing with a linear cost function $g(x)$ as the one considered in (7). The $c\mu$ rule consists in processing the buffers B_{ij} according to a priority established by the product c times μ, where in the present problem, the cost coefficient c associated to the buffer B_{ij} is given by the coefficient c_{ij} in (7) and the maximum processing rate μ for this buffer is given by $\mu_i = \frac{N_c}{T_i}$: this is actually the maximum processing capacity for goods with destination (route) i. There are however two major differences:

- vehicles are not continuous resources;
- the capacity allocation has an influence also on the future (if we allocate all vehicles to destination i we have to wait T_i time units before we can change allocation) while in the scheduling machine case, where the $c\mu$ policy has been proved optimal, capacity allocations can change instantaneously at each step.

4.1 A Possible Heuristic

According to the above observations, we propose here a policy that we believe represents a promising and simple real time rule. We do not give here a proof of optimality for this policy and neither give a proof of stability: the performance of this policy will be explored from a computational point of view. According to the simulations, the stability appears to hold whenever condition (14) holds: this is not surprising since the policy reported below reduces the idle periods with respect to the one considered in the proof of the sufficiency of Theorem 1. This depends on the fact that, even if also this policy (as the one considered in the proof of Theorem 1) does not allow vehicles to travel partially loaded, it is no more required here that all the vehicles travel together to the same destination.

In particular, at each time step, the policy considered in this section allocates the vehicles available at that moment to the buffer B_{ij} which, among the ones with $\sum_j x_{ij} \geq 1$ (that is, among the ones which allow to complete the load of a vehicle) has the largest $c\mu$ index (where, as mentioned above, for the buffer B_{ij}, the index $c\mu$ is given by $c_{ij}N_c/T_i$). To illustrate the policy more in detail, assume for simplicity $Q = 1$ and let $i_1, \ldots, i_P$ be the priority established according to the $c\mu$ rule (that is $c_{i_k}/T_{i_k} \geq c_{i_{k+1}}/T_{i_{k+1}}$ for all k). Then, the policy is given by:

$$n_{i_1}(k) = \min\{N_a(k), \lfloor x_{i_1}(k) \rfloor\}$$

$$n_{i_2}(k) = \min\{N_a(k) - n_{i_1}(k), \lfloor x_{i_2}(k) \rfloor\}$$

and so on, where $N_a(k)$, defined in (3), is the number of available vehicles in the interval (t_k, t_{k+1}). Then, to fill all the vehicles assigned to route i, we set:

$$u_i(k) = n_i(k).$$

4.2 Simulative Results

We tested the policies discussed above in a system with $Q = 1$ (a single product), $P = 3$, characterized by the following parameters: delays $T_1 = 4$, $T_2 = 3$, $T_3 = 5$; arrivals, constant in time, $d_1 = d_2 = 7$, $d_3 = 5$; with this choice the minimum N_c guaranteeing stability is 74, according to condition (14). In figure are shown the performances (6), (7), with unit costs $c_1 = c_3 = 1$, $c_2 = 2$, and $\gamma = 1$, of three policies derived by simulating the system, for a finite time horizon, for various values of the parameter N_c. The dash dotted line shows the performances of the stabilizing policy described in Theorem 1; the dashed line the performances of the policy which allocates at each time instant all the available vehicles prioritizing the buffers with higher content, and the continuous line the performances of the "$c\mu$ policy" (those coefficients, by the parameters chosen, make buffer 2 the one with higher priority followed by buffer 1 and 3).

It is possible to observe that for values of N_c lower than the stabilizing value (74), none of the policy described can achieve stability, consistently with Theorem 1 (for $N_c < 74$ the costs reported in Figure 1 result finite as a consequence of the finite time horizon considered). For $N_c > 74$ the $c\mu$ policy performs better than the policy prioritizing the higher buffers.

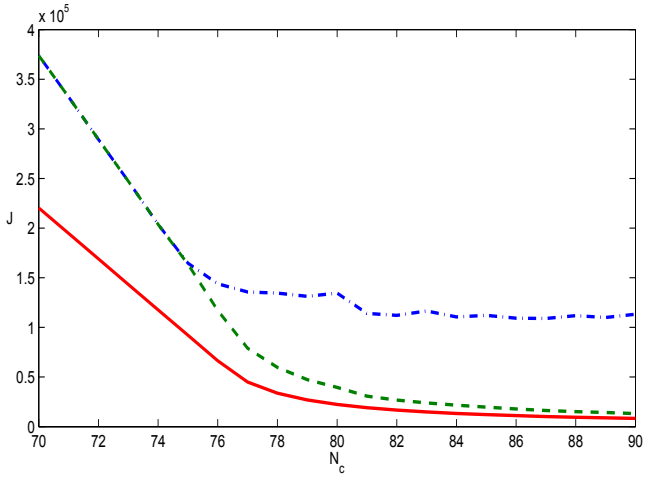


Fig. 1. Performances of the $c\mu$ policy (continuous line), the serve-largest-buffer policy (dashed), and the basic stabilizing policy (dash-dotted), as a function of N_c.

5 Conclusions

In this paper, a simplified model of a logistic node has been considered, where items arrive from outside to the node and must be routed to different destinations. Waiting items are stored in different buffers, according to their class and destination. At first, a necessary and sufficient condition is given in the paper for the possibility of finding dispatching dynamic policies that maintain all the buffers bounded. Subsequently an

optimization problem is considered and a simulative comparison of the performance of different feedback policies is presented in the paper. The problem has been studied under a fluid approximation of the items traveling in the node: this allows to completely fill the vehicles (and to neglect complex combinatorial loading problems). This possibility is actually used by the policies studied in this paper that do not allow the vehicles to travel partially filled. This is actually a reasonable choice under heavy traffic conditions where allowing the possibility of sending partially full vehicles may even compromise the stability, but may become significantly sub-optimal in the case of reduced inflow rates, large holding costs and small traveling costs.

References

1. T. G. Crainic "Long Haul Freight Transportation" Handbook of Transportation Science, R.W. Hall (Ed.), 2nd Edition, Kluwer, 2002
2. Carvalho, T.A. and Powell, W.B. "A Multiplier Adjustment Method for Dynamic Resource Allocation Problems" Transportation Science, 34(2):150-164 (2000).
3. W. Powell, J. Shapiro, and H. Simao. "A Representational Paradigm for Stochastic, Dynamic Resource Transformation Problems" Annals of Operations Research, 104:231-279, 2001
4. Jinghua Xu; Hancock, K. L.; "Enterprise-wide freight simulation in an integrated logistics and transportation system", IEEE Transactions on Intelligent Transportation Systems, Volume 5, Issue 4, Dec. 2004, 342 - 346
5. Powell, W.B. and Carvalho, T.A. "Dynamic Control of Logistics Queueing Networks for Large-Scale Fleet Management", Transportation Science, 32(2):90-109. 1998
6. Powell, W.B. and Carvalho T.A., "Real-Time Optimization of Containers and Flatcars for Intermodal Operations", Transportation Science, Vol. 32, No. 2, 110-126, 1998
7. Yang, J., Jaillet, P. and Mahmassani, H. S., "Real-time multi-vehicle truckload pick-up and delivery problems", Transportation Science, Vol. 38, 135-148, 2004
8. L.C. Dall'Orto, T.G. Crainic, J.E. Leal and W.B. Powell, "The single-node dynamic service scheduling and dispatching problem", European Journal of Operational Research, 170, pp. 123, 2006
9. M. G. Speranza, and W. Ukovich, "Minimizing Transportation and Inventory Costs for Several Products on a Single Link", Operations Research, Vol. 42, No. 5, pp. 879-894, 1994
10. L. Bertazzi, "Analysis of Direct Shipping Policies in an Inventory-Routing Problem with Discrete Shipping Times", Management Science, Vol. 54, No. 4, April 2008, pp. 748762
11. J.R. Perkins and P.R. Kumar, "Stable, Distributed, Real-Time Scheduling of Flexible Manufacturing/Assembly/Disassembly Systems," IEEE Transactions on Automatic Control, Vol. 34, no. 2, February 1989
12. F. Martinelli, C. Shu, and J. Perkins. "On the optimality of myopic production controls for single server, continuous flow manufacturing systems." IEEE Transactions on Automatic Control, 46(8):12691273, 2001
13. P. Chen, Z. Fu, A. Lim, and B. Rodrigues "Port Yard Storage Optimization" IEEE Transactions on Automation Science and Engineering Vol. 1, No. 1, July 2004
14. E. K. Bish, F. Y. Chen, Y. T. Leong, B. L. Nelson, J. W. Cheong Ng, and D. Simchi-Levi "Dispatching vehicles in a mega container terminal" OR Spectrum, 27, 491-506 (2005)
15. S. Kang, J. C. Medina, and Y. Ouyang, "Optimal operations of transportation fleet for unloading activities at container ports" Transportation Research Part B 42 (2008) 970984

A Metamodelling Approach to the Management of Intermodal Transportation Networks

Valentina Boschian[1], Mariagrazia Dotoli[2], Maria Pia Fanti[2]
Giorgio Iacobellis[2] and Walter Ukovich[1]

[1] Department of Electronics, Electrical, Engineering and Computer Science
University of Trieste, Trieste, Italy
valentina.boschian@phd.units.it, walter.ukovich@deei.units.it
[2] Department of Electrical and Electronic Engineering 70125 Bari, Italy
{dotoli, fanti, iacobellis}@deemail.poliba.it

Abstract. The paper develops a novel and broad metamodel of a generic Intermodal Transportation Network (ITN), devoted to provide a reference model for the real time management and control of such systems. In order to take operational decisions, the presented model describes in detail the ITN structure and dynamic evolution that is updated on the basis of the information obtained in real time by modern information and communication technologies tools. The proposed metamodelling approach consists in employing a top down procedure and is based on the UML formalism, a graphic and textual modelling language intended to describe systems from structural and dynamics viewpoints. Hence, the paper models a generic ITN starting from the network description and shows as an example the metamodel of one of the most important nodes that compose it: the port subsystem. To this aim, we present the main UML diagrams describing the structure and dynamics of a case study.

1 Introduction

An Intermodal Transportation Network (ITN) can be defined as a logistically linked system integrating different transportation modes (rail, ocean vessel, truck etc.) to move freight or people from origin to destination in a timely manner [9]. The 21st century will see a renewed focus on ITN, driven by the necessity of moving ever growing quantities of goods and by the technological evolution each of the transportation modes has recently gone through. However, ITN decision making is a very complex process, due to the dynamical and large scale nature of the ITN, the hierarchical structure of decisions, as well as the randomness of various inputs and operations. Typically, ITN management techniques are based on a three-level hierarchy: strategic, tactical and operational ones. Strategic planning involves ITN design and considers time horizons of a few years. Tactical planning basically refers to the optimization of the flow of goods and services through a given ITN. Finally, operational planning is short-range planning and involves transportation scheduling at all transporters on an hour-to-hour basis, subject to the changing market conditions as well as to unforeseen transportation requests and accidents.

The related literature has largely addressed the ITN modelling and management problems at strategic and tactical levels. Recently, entrepreneurs and researchers are being attracted by the key problem of using effectively and efficiently the latest developments of ICT (Information and Communication Technologies) for ITN operational management [6], [11], [12], [13]. In fact, since intermodal transportation is more data-intensive than conventional transportation, ICT are considered a primary "enabling tool" for the safe and efficient real time management and operation of ITN. Indeed, today the effective use of the modern ICT tools has made it possible to know the state of the system in real time and therefore manage and change on-line paths, vehicle flows, deliveries and orders. In order to operate such choices, there is a need of dynamic models that can track the state changes of the various system components and determine operation indices typical of the real time management, such as utilization, traffic indices and delivery delays [14]. In the domain of ITN models at the operational level we recall the class of discrete event system models [1], [3], [4] and of the simulation models [13], [14]. However, the above models are designed to describe a particular ITN and do not fully take into account the multiplicity of elements that can influence the ITN dynamics and the corresponding information structure. Since ITN are often complex and distributed systems, they have also been effectively represented by multi-agent techniques. However, this promising approach to transportation and traffic management is still at its early stages [2].

This paper develops a novel metamodel of ITN at the operational level intended for real time management and control of such systems. The metamodel has a general and modular structure and is characterized by high information integration. In order to take operational decisions, the presented model describes in adequate detail the structure and dynamic evolution of the ITN and is updated on the basis of data exchanged by the players in the system and information obtained in real time by using modern ICT tools [7]. The proposed approach consists in applying metamodelling by a top down procedure based on the UML formalism [9], [10], a graphic and textual modelling language intended to understand and describe systems from various viewpoints. Hence, UML enables us to describe the structure and dynamics of a generic ITN starting from the description of the network, until the model of the most important entities that compose it (classes) and their corresponding activities. Moreover, UML unifies the formalism by using appropriate and effective diagrams that can be easily translated into a simulation software. Indeed, the approach to the management and control of ITN is based on the construction of a reference model that simulates the evolution and dynamics of ITN and provides to the control modules the knowledge base necessary for decisions in real time. To this aim, the model reproduces the behaviour of the ITN by storing the real events such as variations in demand and orders, blockages, accidents, breaks and all those occurrences that interact with the flow and management of materials and transporters. Hence, based on the knowledge of the reference model state and the events, decision makers can make the appropriate choices optimizing suitable performance indices. To the best of the authors' knowledge, such a UML based metamodel approach has never before been proposed for ITN.

The paper is organized as follows. Section 2 describes the main steps of the ITN metamodel approach. Subsequently, sections 3 and 4 respectively describe the static and dynamic diagrams in the ITN metamodel. A conclusion section closes the paper.

2 The Metamodel of Intermodal Transportation Networks

We consider a generic ITN constituted by a set of terminals (ports, airports, railway stations, etc.), together with the interconnections between them and land infrastructures. Metamodelling is a technique that applies to models [5]. A metamodel provides an accurate description of the constructs and rules needed to obtain semantic models and encapsulates all the concepts that are necessary to describe the structure and dynamics of a particular system.

The metamodelling approach presented in this paper is a top-down approach that decomposes the system to gain insight into its compositional sub-systems. In the top-down approach an overview of the ITN is formulated, specifying any first-level subsystems: ports, airports, railway stations, intermodal terminals, ground, sea and air connections, information systems, carrier and freight forwarder. Subsequently, each subsystem is refined in detail from the structural and dynamical points of view, using suitable UML diagrams, such as class diagrams and activity diagrams. In the following sections we detail the main steps of the proposed ITN metamodelling approach. First, a procedure addressing static models is devised, defining the so-called package diagram and the class diagrams. Subsequently, process flows are considered and dynamic models are obtained, by referring to the so-called activity diagrams.

3 Static Modelling of Intermodal Transportation Networks

3.1 The Package Diagram

The first step of the presented metamodelling approach consists in identifying the main subsystems composing an ITN. These can be divided into structural subsystems (i.e., ports, airports, railway stations, intermodal terminals, ground, sea and air connections, carrier and freight forwarder) and the information system. They represent the generic concepts used within the metamodelling framework and are modelled in UML by *packages*. However, such subsystems are complex nodes that can be considered composed by other generic objects (or classes). Hence, we represent the overall ITN by the UML package diagram [9], [10]. Packages are groups of entities related to each other. Figure 1 depicts the package diagram of a generic ITN. We identify the following seven packages that form the ITN: the port, the airport, the railway station, the ground, sea and air connection, the intermodal terminal, the information system and the carrier and freight forwarder (see Fig. 1). Each package is composed by different classes representing structural basic objects interconnected with each other. Arrows show the cases in which a class in one package needs to use a class in another package. This causes a dependency between packages: for example, the information system is updated on the basis of data obtained in real time using modern ICT tools.
We assume that each package includes an information class representing the informative structure devoted to manage the considered system. However, we consider also the possibility of the existence of a centralized information system that can manage and coordinate different packages. For example, the port package contains an information class that manages the flow of trucks, trains, cranes, etc. On the other hand,

the external and higher level information system can control the interactions between the port and the infrastructures, by receiving data from the port area and the ground, sea, rail and air connections.

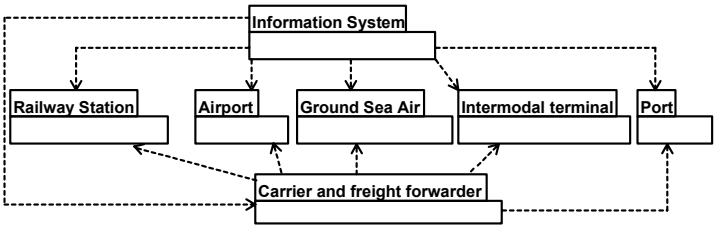


Fig. 1. The package diagram of the ITN: arrows show dependence among packages.

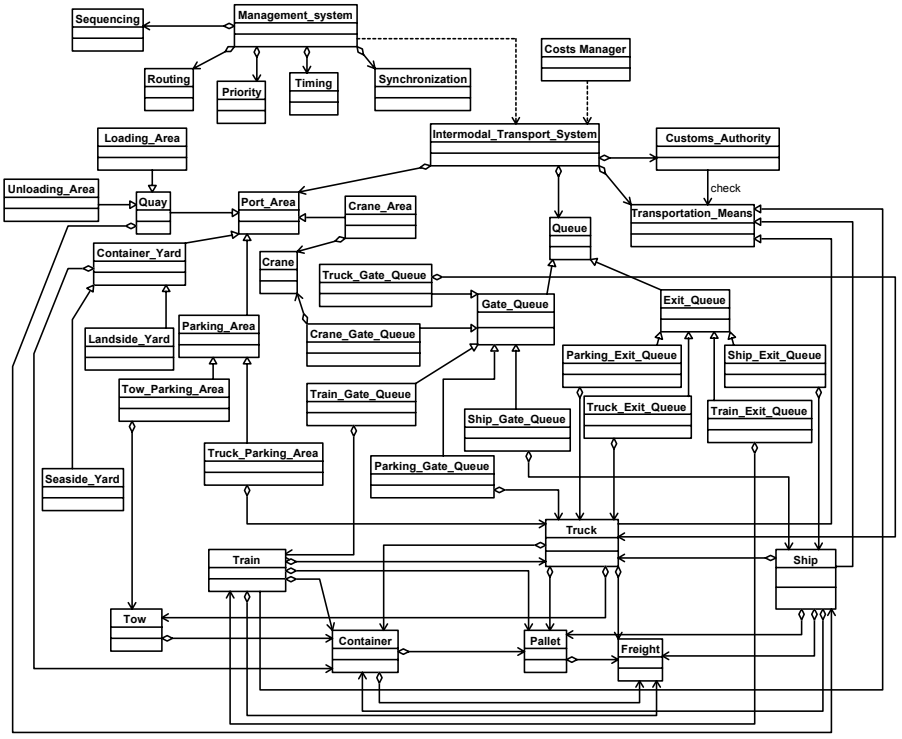


Fig. 2. The port class diagram: connections show relationships among objects in the class.

3.2 The Class Diagrams

The subsequent step of the static modelling consists in setting up the so-called class diagrams, specifying the configuration of the various packages defined in the previously discussed package diagram [9], [10]. A class diagram describes the types of objects in the ITN and the various static relationships between them. These are represented by different pictorial connections and may be relationships of association

(solid line), aggregation (solid line with a clear diamond at one end), composition (solid line with a filled diamond at one end), inheritance or generalization (solid line with a clear triangle at one end), realization (dashed line with a clear triangle at one end) and dependency (dashed line with an arrow at one end). Moreover, the class diagram may show the features of a class, i.e. the name, attributes and operations of the class.

As an example, we show in Fig. 2 the class diagram of one of the packages in Fig. 1, namely the Port: for the sake of brevity we omit the class diagrams of the remaining packages in Fig.1, since they may be set-up similarly to the diagram described here. The main classes included in the diagram in Fig. 2 are the Intermodal_Transport_System, the Management_System and the Cost_Manager. In the sequel we briefly describe such classes together with their typical attributes and methods. Obviously, these features may vary depending on the ITN under study [14].

The Intermodal_Transport_System class represents the overall port terminal system. The class attributes are: 1) the dynamic lists of ships, trains and trucks currently in the terminal; 2) the dynamic lists of ships, trains and trucks already served by the operators in the terminal or by the quay cranes, waiting for permission to exit from the terminal; 3) the dynamic lists of ships, trains and trucks queued and waiting for service; 4) the dynamic lists of ships, trains and trucks currently being served; 5) the dynamic lists of ships, trains and trucks currently leaving the terminal; 6) the lists of occupied quay cranes and available ones. The class methods are: 1) the registration of ships, trains and trucks entering the terminal; 2) the extraction from the list of ships, trains and trucks waiting for service; 3) the extraction from the list of available cranes; 4) the assignment of a crane to a specific task of freight loading/unloading; 5) the crane activation; 6) the extraction from the list of ships, trains and trucks exiting from the terminal; 7) the update of the list of served ships, trains and trucks; 8) the update of the list of waiting ships, trains and trucks; 9) the update of the list of ships, trains and trucks exiting the terminal; 10) the update of the list of available cranes. The Intermodal_Transport_System class aggregates the following classes: Port_Area, Queue, Transportation_Means and Customs_Authority.

The Port_Area class represents the physical port area, modelled generalizing the following classes: Quay, where the ship loading/unloading processes take place, Container_Yard, representing the freight stock area, Crane_Area, describing the crane pick up and delivery point and Parking_Area, representing the zones where vehicles are parked. Hence, all these classes inherit the properties and basic services of their parent class, i.e. the Port_Area. This has the following attributes: 1) dimensions; 2) list of occupied locations; 3) list of unoccupied locations; 4) opening time; 5) closing time; 6) number of operators. Its methods are: 1) the extraction from a list of locations in the port area; 2) the access control; 3) the assignment of a location to an entity (ship, train, truck); 4) the clearing of a location upon the leaving of an entity. The Container_Yard class defines the storage area where freight is stored and waits for being delivered to their destination. The Quay class models the quay where ships are loaded/unloaded. Hence, the Quay class aggregates the Ship class, while it generalizes the Loading_Area and Unloading_Area classes. The Crane class attributes are: 1) the crane type (quay crane loading/unloading freight onto/from the ship, stacker crane retrieving/storing freight from the Container_Yard, automated guided vehicle or tractor moving freight from the quay area to the container yard area, etc.); 2) the crane

identification number. The Loading_Area class models the ship loading area, while, the Unloading_Area class represents the ships unloading area. The Parking_Area class is the area where trucks wait either to be loaded or to be embarked. The class also includes means of transportation waiting for any reason. The Parking_Area class obviously includes the Truck class and generalizes the Truck_Parking_Area and Tow_Parking_Area classes. The Container_Yard class includes the Container class and generalizes the Seaside_Yard and Landside_Yard classes. The Seaside_Yard (Landside_Yard) class represents the warehouse area where containers, freight, tow or pallets to load on trains or trucks destined to the inland (to be shipped) are stored.

The Queue class enables the management of queues and the computation of waiting times and hence of costs. The attributes of this class are: 1) the maximum number of entities a queue may have; 2) the queue management policy; 3) the velocity of queue management; 4) the costs associated to the waiting time in queue of a user; 5) the queue identification number; 6) the current number of users in a queue; 7) a Boolean flag indicating whether the queue is full. Its methods are: 1) the inclusion of an entity in the queue; 2) the cancellation of an entity from the queue; 3) the queue management; 4) the queue cost computation. Its children classes are called Gate_Queue and Exit_Queue, respectively representing the input and output queues in all the port, e.g. queues in parking area, in crane area, etc. The Gate_Queue and Exit_Queue both have as children classes those representing the truck, train, ship, parking and crane queues. For instance, the Parking_Gate_Queue and the Parking_Exit_Queue classes respectively represent the input and output truck queues in the parking area.

The Transportation_Means class represents the transportation means circulating in the port, i.e., trucks, trains and ships. Hence it generalizes the Truck, Train and Ship classes and is associated to the Customs_Authority class. Indeed, customs have the task of controlling the arriving transportation means and applying the corresponding taxes. The Transportation_Means class exhibits the attributes: 1) the transportation means identification number; 2) the transportation means dimensions; 3) the carrier name; 4) the category of transported goods (e.g., hazmat); 5) the goods place of origin; 6) the goods destination; 7) the goods weight; 8) the identification number of the carried tows, containers, freight or pallet; 9) a Boolean flag indicating full load; 10) the current waiting time for the unload and load operation. The class methods are as follows: 1) the load/unload operation; 2) the waiting time computation. Note that the class diagram shows that the Truck class is included in the Train and Ship classes, while it aggregates the Tow, Container, Freight and Pallet classes. The Container class includes the Freight and Pallet classes, which in turn includes the Freight class. Note that both the Tow and Container class have one attribute, the identification number. On the contrary, the Pallet class attributes are the identification number and its capacity, while the Freight class attributes are the identification number and dimension. The Train class represents trains moving freight entering or exiting from the port. Such a class includes the Container class and makes use of the Intermodal_Transport_System class, so that trains remain in the railway as minimum time as possible. The Ship class models ships berthed at the port. This class includes the Container and Truck classes.

The Customs_Authority class represents the customs and their tasks of controlling all transportation means and their freight and applying taxes. It is therefore connected to the Transportation_Means class by an association function named “check”. Its

attributes are: 1) the opening and closing times; 2) the control time of goods; 3) the number of customs operators. Its methods are: 1) the freight control; 2) the tax assignment.

The Management_System class is devoted to managing the whole Intermodal_Transport_System class. This class provides the rules to perform choices and decisions in the ITN. Hence, it *uses* the database provided by the Intermodal_Transport_System class to elaborate the decisions. It *aggregates* the following classes: the Synchronization class, introduced to minimize operation delays in the intermodal terminal so as to maximize synchronization of operations; the Priority class, which deals with the assignment of priorities to the transportation means (e.g., trucks transporting hazmat are assigned priority or a FIFO logic may be considered); the Routing class, which describes the operation path that has to be followed; the Sequencing class, assigning the subsequent operation that the system has to execute; the Timing class, which determines the timing of the next operation.

Finally, the diagram in Fig. 2 includes the Costs_Manager class that computes the managing costs of the container terminal. Hence, this class uses data stored in the Intermodal_Transport_System and exhibits the cost calculation operations.

4 Dynamic Modelling of Intermodal Transportation Networks

In this section we employ activity diagrams to provide a "dynamic view" of the system. Activity diagrams aim to describe the logic of the involved processes and the workflow [9], [10]. They are similar to flowcharts, but they allow representations of parallel elaborations in order to explain the critical points in the processes and workflow of the whole system by pointing out all the possible paths, parallel activities and their subdivisions. The main elements of these diagrams are: the initial activity (denoted by a solid circle); the final activity (denoted by a bull's eye symbol); activities, represented by a rectangle with rounded edges; arcs, representing flows, connecting activities; forks and joins, depicted by a horizontal split, used for representing concurrent activities and actions respectively beginning and ending at the same time; decisions, representing alternative flows and depicted by a diamond, with options written on either sides of the arrows emerging from the diamond; swim lanes, highlighting responsibilities; signals representing activities sending or receiving a message, which can be of two types: input signals (message receiving activities), shown by a concave polygon, and output signals (message sending activities), shown by a convex polygon. Moreover, activities may involve different participants in a system. Hence, partitions are used to show which actor is responsible for which actions and divide the diagram into columns or swim lanes.

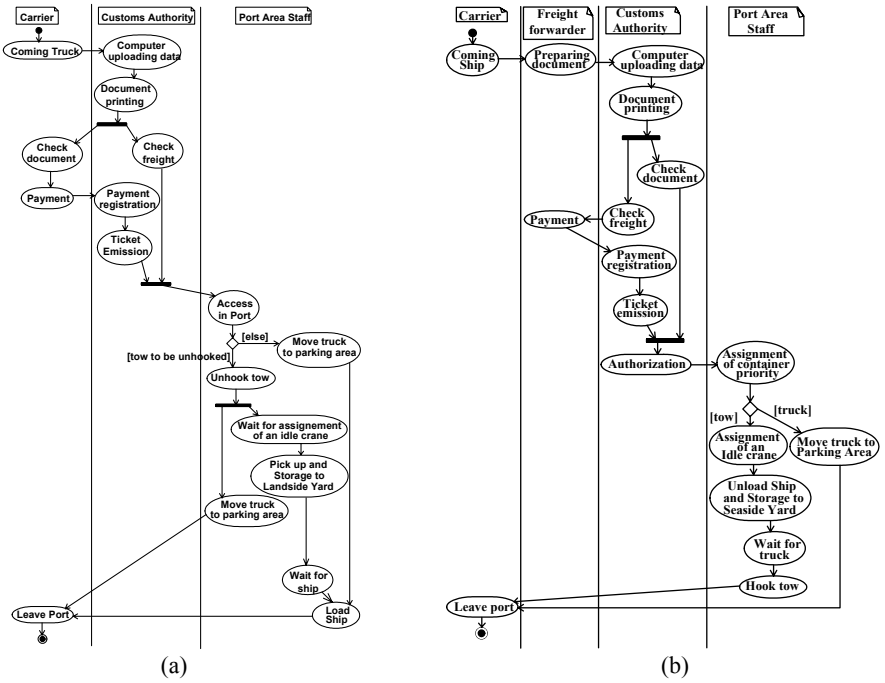


Fig. 3. The activity diagrams of the ship loading (a) and unloading (b) processes.

In the following we describe the activity diagrams of two basic port operations: the ship loading and unloading processes. Note that, in order to illustrate in detail the workflow of these processes, we refer to the case study of the Trieste port (Italy) [1].

4.1 The Activity Diagram of the Ship Loading Process

The logic flow associated to the ship loading operations of the port of Trieste [1] is described by the activity diagram depicted in Fig. 3a. The actors of these activities are the carrier (included in the Carrier and Freight Forwarder package, see Fig. 1), the customs authority (represented by the corresponding Customs_Authority class in the class diagram of Fig. 2) and the port area staff (operators that are attributes of the Port_area class). When trucks arrive at the port area, the loading documents are prepared by the carrier. They contain data about the freight producer (e.g., company name, country of origin etc.), the transported goods (e.g., quality, quantity etc.) and the vehicle (e.g., nationality, number plate etc.). These documents are uploaded in electronic form and printed by the customs authority. The printed documents are checked by the carrier and the freight is checked by the customs authority. Shipping tariffs are paid by the carrier. Subsequently, the payment is registered and the ticket is printed by the customs authority.

After the ticket emission and freight check, the port area staff has to manage the access of trucks into the port. We enlighten that in the considered port terminal the trucks arriving at the port can either be loaded into the ship with their tow or the tow

can be unhooked and successively it is loaded into the ship [1]. At this point, on the basis of suitable priority rules that are established by the management system, either a complete truck can be loaded into the ship or a tow is unhooked and waits for the assignment of an idle crane. More precisely, trucks move to the parking area and can subsequently be loaded onto the ship. Alternatively, after the unhooking process, tows are loaded on a crane and are stored in the landside yard were they wait for the ship loading. Finally, tows are loaded by the quay crane on the ship.

We remark that the described logic flows may be significantly improved by suitable arrangements employing modern ICT tools, e.g. setting up a truck tracking device, an electronic transportation document system, etc. Obviously, the corresponding activity diagram can be obtained by suitably updating the ship loading process depicted in the diagram of Fig. 3a.

4.2 The Activity Diagram of the Ship Unloading Process

The logic flow associated to the ship unloading operations is similar to the previously analyzed loading phase and is described as follows (see Fig. 3b). The ship enters the port and the freight forwarder prepares the documents to unload vehicles and goods. These documents are uploaded in electronic form and printed by the customs authority. Data contained in the documents and freight are checked. Tariffs to unload the ship are then paid by the freight forwarder, successively the payment is registered and the ticket is printed. When the documents are checked and the tickets are ready, the authorization is issued. Hence, the port area staff, on the basis of the priority assigned by the management system class either enables the exit of trucks or assigns an idle crane to tows. In the latter case the tows are parked in the seaside yard, where they wait for trucks. Finally, trucks hook the tows and leave the port.

Similarly to the activity diagram of the ship loading process, also the activity diagram of the unloading process may be significantly improved by modern ICT tools.

5 Conclusions

The paper presents a novel top down procedure to develop a metamodel of Intermodal Transportation Networks (ITN), devoted to real time management and control at the operational level. To this aim, the model describes in detail the structure and dynamic evolution of the ITN so that it can be updated in real time using information from the real system obtained by modern information and communication technologies tools. The proposed metamodelling procedure is based on the UML formalism, a graphic and textual language able to describe systems from structural and dynamics viewpoints. To show the proposed model effectiveness, the paper focuses on a particular node of the ITN, i.e. the port, with particular reference to the port of Trieste (Italy). The detailed description of the main system components and of two basic processes of the port show how the UML diagrams can effectively depict the structure and activities of such complex and large systems. Hence, the proposed metamodelling approach and the used UML formalism provide a reference model that closely

simulates the evolution of the ITN. This feature of the reference model is crucial and can be effectively employed in order to supply the control modules with the knowledge base necessary for decisions in real time. Future research will address the model of all the nodes of the ITN and the specification of the decision and control modules. In addition, we plan to apply and validate the proposed approach to a real case study. To this aim, preliminary studies are being carried out with several European ports and authorities in the framework of a research project funded by the European Commission. Finally, further studies could be developed to support learning from data in the provided model, e.g. adding decision support and control modules based for instance on agent techniques.

References

1. Danielis, R., Dotoli, M., Fanti, M.P., Mangini, A.M., Pesenti, R., Stecco, G., Ukovich, W.: Integrating ICT into Logistics Intermodal Systems: a Petri Net Model of the Trieste Port. Proc. Eur. Control Conf., Budapest Hungary (2009).
2. Davidsson P., Henesey L.E., Ramstedt E., Törnquist J., Wernstedt F.: An Analysis of Agent-Based Approaches to Transport Logistics. Transportation Research Part C: Emerging Technologies, Vol. 13 (2005), 255-271.
3. Di Febbraro, A., Porta, G., Sacco, N.: A Petri Net Modelling Approach of Intermodal Terminals Based on Metrocargo System. Proc. Intelligent Transportation Systems Conf. (2006), 1442–1447.
4. Fischer, M., Kemper, P.: Modeling and Analysis of a Freight Terminal with Stochastic Petri Nets. Proc. 9th IFAC Symposium Control in Transportation Systems (2000).
5. Ghazel M., Toguyéni A., Bigand M.: An UML Approach for the Metamodelling of Automated Production Systems for Monitoring Purpose. Computers in Industry, Vol. 55 (2004), 283-299.
6. Giannopoulos, G.A.: The Application of Information and Communication Technologies in Transport. Eur. J. Oper. Res., Vol. 152 (2004), 302-320.
7. Ikkai, Y., Oka, H., Komoda, N., Itsuki, R.: An Autonomous Distributed Information System for Logistics Control with Data Carriers. 2003 IEEE Int. Conf. Industrial Technology (2003), 1079-1082.
8. Macharis, C., Bontekoning, Y.M.: Opportunities for OR in Intermodal Freight Transport Research: a Review. Eur. J. Oper. Res., Vol. 153 (2004), 400-416.
9. Miles, R., Hamilton, K.: Learning UML 2.0. O'Reilly Media, Sabastopol CA USA (2006).
10. OMG: UML Resource Page. Available at http://www.uml.org/ (2009).
11. Ramstedt, L., Woxeni,us J.: Modelling Approaches to Operational Decision-Making in Freight Transport Chains. Proc. 18th NOFOMA Conf. (2006).
12. Verbraeck, A., Versteegt, C.: Logistic Control for Fully Automated Large-Scale Freight Transport Systems. Proc. IEEE Conf. Intelligent Transportation Systems (2001), 770-775.
13. Xu, J., Hancock, K.L.: Enterprise-Wide Freight Simulation in an Integrated Logistics and Transportation System. IEEE Trans. Int. Transp. Sys., Vol. 5 (2004), 342-346.
14. Yun, W.Y., Choi, Y.S.: A Simulation Model for Container-Terminal Operation Analysis Using an Object-Oriented Approach. Int. J. Prod. Econ., Vol. 59 (1999), 221-230.

System Interface of an Integrated ISS for Vehicle Application

M. A. Hannan, A. Hussain and S. A. Samad

Department of Electrical, Electronic and Systems Engineering
Faculty of Engineering and Built Environment, National University of Malaysia
43600 Bangi, Selangor, Malaysia
{hannan,aini,salina}@eng.ukm.my

Abstract. This paper deals with the system interface of the integrated intelligent safety system that involves the vehicle applications of airbag deployment decision system (ADDS) and tire pressure monitoring system (TPMS). Lab Window/CVI in C interface program is developed for prototype implementation. The prototype implementation is the interconnection between the hardware objects of the intelligent safety devices such as load cell, Logitech web-camera, Cross-bow accelerometer, TPM module and receiver module, data acquisition card, CPU card and touch screen. Integration of several safety subsystems decision such as the image processing module, weight sensing module and the crash detection module are fused for intelligent ADDS. The integrated safety system also monitors the tire pressure and temperature and analyze the TPMS data for tire condition classification. The system interface developed the integrated system prototype performance that is evaluated through several test runs. Their result proven the embedded intelligent safety system is unique, robust and intelligent for vehicle safety application.

1 Introduction

The term "intelligent safety" does not literally mean that intelligence resides inside the vehicle. The "intelligence" implies here as an active essential part of the vehicle that contribute to safety, security and driving comfort. Intelligent safety system provides operation to ensure the safety and comfort level of the occupant in the vehicle [1]. However, due to occupant expectation of high level of control and safety, a large number of individual safety systems are needed [2]. This increased concern for safety issues has resulted to design accurate integrated intelligent safety system that involves features technologies, characteristic of safety issues and providing solutions by monitoring, detecting, classifying, impending crash or unsafe driving conditions, warning the driver, improving his or her ability to control the vehicle and prevent the accident [3].

Many individual researches have been focused on the safety issues such as occupant detection, classification and position, vehicle crash detection and its severity analysis, TPMS etc. For example, occupant detection and characterization are the fundamental importance [4] to improve the safety and comfort features of the

occupants. However, it is a difficult task, despite the success of some of these systems, the occupant detection such as human, non-human object and its classification still pose a number of challenges with respect to real time implementation and operation [5].

The vehicle crash detection is very helpful for preventative safety, preventing accidents and collision for minimizing human injury when an accident occurs [6]. However, in past it has been a rather seldom discussed on research and theoretical analysis of crash in the field of traditional engineering [7]. It is therefore NHTSA and other safety concern made rule that vehicle crash detection and analysis are mandatory for safety issue [8].

Similarly, the safety issue on TPMS is a significant factor in the driving experience and vehicle performance [9]. Accordingly, the NHTSA made a legislation known as the TREAD act in which after 31 October 2006, all vehicles in United States should have the option of TPMS [10].

The system interface for the prototype implementation of intelligent safety system is the most important. It is therefore, this paper pursue the interface program that are used for the development of an innovative integrated intelligent safety system to identify major hazards and assess the associated risks in an acceptable way in various environments where such traditional tools cannot be effectively or efficiently applied. The safety device provides data to the intelligent safety system that is useful for the development of ADDS and TPMS. The objective of this paper are met by integrating and developing the advance solution of the innovative safety issues likewise occupant detection, classification and position, vehicle crash detection and its severity analysis, TPMS and other detrimental issues.

2 System Integration

There are a number of limitations with the conventional architecture of the vehicles intelligent safety systems. The safety measures are challenging and increasing awareness of the automotive companies, due to customer expectation of high level of control and safety. However, because of the wide variety of individual safety system, it needs a large library of programs and much expensive individual platforms. This wide variety of safety is continuous increasing the complexity that would lead to a physical maximum and interdependence between the systems. So as to extract the most of the individual demands, the platform need to be integrated, addressed a robust algorithm and calibration process to optimized and validate the vehicle integrated safety system. The principle motivation behind the system integration is to reduce individual systems safety device management cost in the performance domain. The integrated intelligent safety system aims to provide heterogeneous workload management concepts and functions to the safety issues and validate them based on performance diagnosis of collected monitoring data in a developed platform.

The intelligent vehicle safety platform identifies the hardware and software execution environment of a system. The hardware platform identifies a set of hardware objects associated with processors. The system interface provides a high level of interface between software objects running on different processors that control the hardware. The proposed integrated safety system deals with the safety and

comfort issues in the modern vehicle such as TPMS, occupant detection, detection and position and vehicle crash detection. This integrated safety system gathers data through a set of sensors, collected the data through acquisition processes and eventually reacts through a CPU and finally, safety issues are monitored in a LCD display unit.

3 Method and Algorithm

We have developed the algorithms for ADDS and TPMS. In ADDS, we have developed the individual algorithms for occupant detection, classification and position based on weight sensing and image process as well as for vehicle crash detection. In weight sensing, in order to classify, the weight measurement data are used with logic combination. We consider that less than 10 kg as a non-human object, while the child setting is '10 kg<child<35' and the adult setting is 35kg<adult<100kg. For example, when an adult occupant is on the seat, the adult logic is true, and child and non-human object logics are false, which the dynamic output classifies as adult and displays its decision on the monitor. In position detection, we have calculated the centroidal distance Fx and Fy are as follows.

$$Fx = x\frac{(-F1 + F2 - F3 + F4)}{(F1 + F2 + F3 + F4)} \tag{1}$$

$$Fy = y\frac{(F1 + F2 - F3 - F4)}{(F1 + F2 + F3 + F4)} \tag{2}$$

where F1, F2, F3 and F4 are weight forces of the four sensors, while x and y are the distances from the centre to the sensor in x and y directions, respectively. These calculations of Fx and Fy provide the appropriate position of the occupant.

In image processing, the algorithmic approach for detection and classification of occupant, non-human object and non-object as shown in Fig. 1. The proposed system is functioned with the combination of fast neural network (FNN) and classical neural network (CNN), in which the FNN is trying to extract any positive detection including false detection. Post-processing strategies are applied to converts normalized outputs and for adjusting intensity histogram equalization or lighting correction function are also applied to solve false detection. The output of FNN is then feed to CNN to verify which region is indeed the system detection. This proposed combined network is quite robust on detecting accuracy and computation efficiency rather than single network, which is unable to fully eliminate false detection problem.

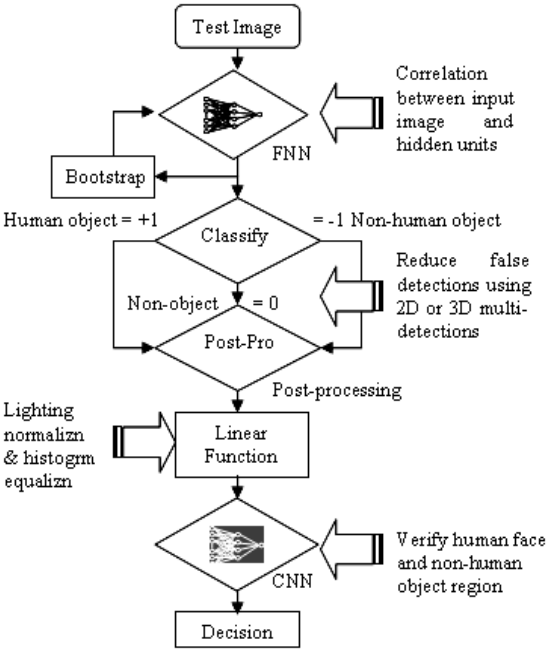


Fig. 1. Neural network algorithm for the occupancy detection.

This variable, Δv(t) is an essential parameter for crash detection that can be found in the algorithms development. The change in vehicle velocity, Δv(t) is obtained by computing the integration of the acceleration signal [hannan elsevier] as shown in below.

$$\Delta v(t) = \int a(t)dt = -A\omega^2 \int \cos(\omega t + \delta) \tag{3}$$

Selecting a threshold value of the vehicle velocity, V_{th}, is therefore required to facilitate the decision making of whether or not an effective crash has occurred. Such threshold V_{th} value can easily be determined from the lowest speed of an effective crash defined by NHTSA i.e. 22.54 km/h. In order to detect crash, the developed algorithmic steps are as follows,

i) If $\Delta v(t) \geq V_{th}$, then output ='1'; DECISION: Effective crash is detected.

ii) If $\Delta v(t) < V_{th}$, then output ='0'; DECISION: Effective crash is not detected.

An increase in vehicle speed during crash increases the crash severity factor. The change of velocity, Δv(t), over a period of time, T, at the detection state can be computed since the integral over the noise component is approximately zero. The circuit for computing Δv(t) can be designed using systolic architecture to achieve the real-time speed. The output of the detection state is fed into a data acquisition card for system development.

In TPMS methodology, there are two ways to acquire data from the sensor likewise, using a successive approximation algorithm or by a threshold check. A successive approximation provides an accurate conversion of the sampled temperature

or pressure reading into an 8-bit value. In the threshold check, the DAR is preloaded with a threshold value during standby/reset mode to detect whether the pressure or temperature has crossed a particular level. The receiver module is capable of receiving both OOK and FSK through UHF receiver that communicates with the CPU by a SPI. The UHF receiver detects and demodulate the signal through Manchester-encoded bit stream, sending out the important data to the CPU that is monitor in the display unit. The TPM and receiver module is loaded with a simple software program to effective the functionality of the hardware. The assemble code for TPM tire module is written using the "WIN IDE" integrated development environment and programmed into RF2 through the programmer board that transmitted data to the receiver module. Receiver module communicates with the UHF receiver using the Turbo C Borland C compiler run under DOS. A function "TPMReceiverModule" created in the main interface program is called in UKM.dll to monitor pressure and temperature that is extracted from TPM receiver through SPI connection to CPU.

4 Prototype Structure

The hardware prototype is a vital representation of final design of integrated safety system. It is also the basic tool to find deep bugs in the hardware. This is why; it has a crucial step in the whole hardware structure design of embedded system.

This system implementation is developed through the physical interconnections between hardware objects using standard hardware design technique. The whole developed system structure consists of the following hardware objects such as sensors as tire pressure monitoring modules, load cell weight sensor, Logitech web-camera (WC) and Cross-bow accelerometer (AC) crash sensor, data acquisition card for analog to digital conversion, CPU card, touch screen for deploying result and ATX switch mode power supply (SMPS) as shown in Fig. 2.

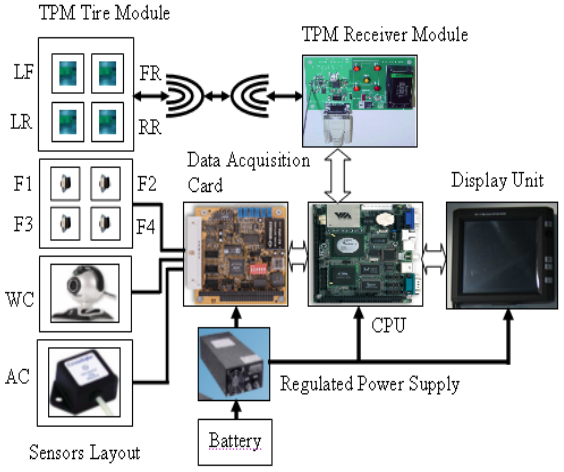


Fig. 2. Hardware structure of integrated prototype system.

5 System Interface

In this interface program, data are acquired from the weight sensor inside the passenger seat and crash accelerometer sensor fixed on the vehicle bumper through AXIOMTEK AX10410A acquisition card. This weight and crash sensor provides analog signal that received by CH0 to CH6 of the A/D converter of DAQ card from the 4 weight sensors and a 3 axis of crash sensor. A web-camera is connected to the CPU through USB. The program firstly determined whether the seat is empty, human or non-human something is on it according to the logic combination of measured weight sense and image on the seat. Then classify the occupant size based on weight measurement data with logic combination. For example, when an adult occupant is on the seat, the adult logic is true, and child and non-human object logics are false, which the dynamic output classifies as adult and displays its decision on the monitor. The interface system also determined the centriodal position of the occupant to find whether the occupant is in good position or not. Occupant detection, classification and position along with vehicle crash detection decision provides decision to the ADDS. TPM receiver module is connected to the CPU through SPI that extracted pressure and temperature for monitoring and provide warning system to the display unit. Details interfacing connection among the external sensors and power supply to the data acquisition card and CPU are shown in Table 1 and Table 2.

Table 1. Connection the external sensor to the DAQ and CPU.

DAQ and CPU	Sensor Type	Wire Color
Channel 0	Weight Sensor	Blue (Left Bottom)
Channel 1	Weight Sensor	Violet (Left Top)
Channel 2	Weight Sensor	Orange (Right Top)
Channel 3	Weight Sensor	White(Right Bottom)
Channel 4	Crash Sensor	Violet (X-axis)
Channel 5	Crash Sensor	Yellow (Y-axis)
Channel 6	Crash Sensor	Black (Z-axis)
USB	Web-Camera	USB Port
SPI	TPM Receiver Module	Serial Port

Table 2. Power connector on the power supply and CPU card.

Wire Colour	Wire Type
Yellow	12 volt
Red	5 volt
Black	Ground

The displays are shown on a touch screen. The touch screen has three different connectors likewise USB, VGA and 12 volt regulated power supply.

Table 3. Touch screen display unit connectors.

Connector on the Touch Screen	Connection to the Other End
USB connector	Connected to the USB connector on the CPU card.
VGA connector	Connected to the VGA connector on the CPU card.
Power connector, Red wire – 12 Volt Black wire – Ground	Red wire goes to the Yellow connector on the SMPS and the Black wire goes to the black connector on the SMPS.

6 Interface Program

The system interface between the software and hardware is developed based on Lab Window/CVI in C programming language. The Low level driver called "c:\cvinterface\UKM.dll" is written as a Win32 DLL file where the functions inside the DLL are called by the Lab Window/CVI C program. In this DLL file, the function called "Func1" processes the analog signal received by CH0 to CH6 of the A/D converter of DAQ card from the 4 weight sensors and 3 axis of crash sensor. The function "HumanDetection" provides the decision based on weight sensing whether the occupant is adult, child or empty. The function "ImageProcess" is called inside the UKM.dll to perform the face detection. This function returns a 1 if the image captured by the web-cam is detected as "human" else if it detects a "Non Human" the function returns a 0.

This 1 and 0 is fused with the logic combination of weight sensor to detect occupant as adult, child, non-human object or empty. The function "CrashSensor" is responsible for whether crash is generated or not. The function for position detection "PositionDetection" calculates the centroidal distance of x and y axis from the UKM.dll that display is the GUI and provides the decision for occupant position. Finally, the function "ABagParm" provides the airbag deployment decision upon fusing logic combination of occupant classification, position and vehicle crash detection decision. The function "TPMReceiverModule" also called in UKM.dll to monitor pressure and temperature that is extracted from TPM receiver through SPI connection to CPU. The Fig. 3 shows the details program flowchart diagram of the UKM.dll.

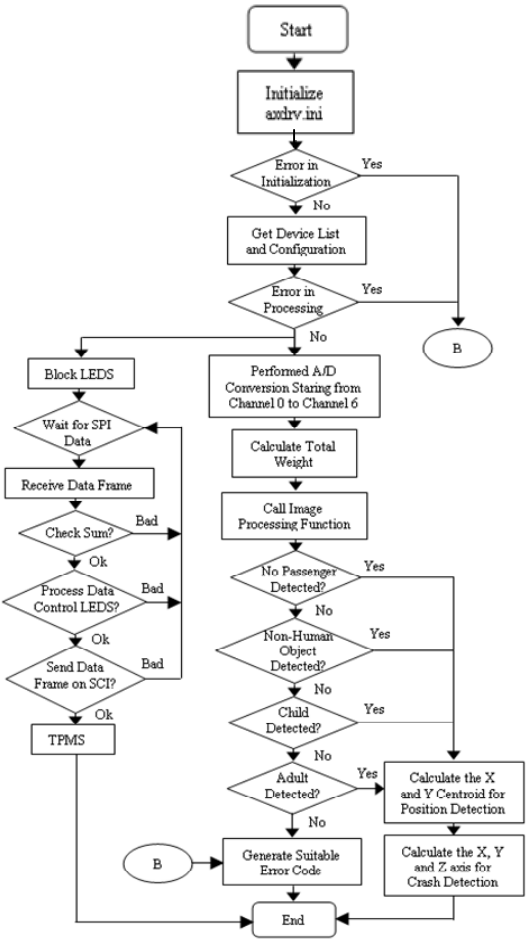


Fig. 3. Interface program flowchart of the integrated system.

7 Experimental Results and Discussion

The experimental results and graphical user interface on the touch screen graphs representing real-time field applications of the integrated intelligent safety system. The results contain for network interface processing, image and signal processing for occupant detection, classification and position, vehicle crash detection and its severity analysis towards ADDS and the performance monitoring of TPMS. Usually the real-time constraints vary up to 1 minute. However, in our prototyped hardware, the execution vectors of whole system are derived from the experimental measurements within 50 ms. The results in decision and graph on the GUI, the resource library is assumed that contain CPU, data acquisition card, sensors and their interface between sensor and CPU. Details of experiment results and its analytical discussions are given below.

7.1 Results and Discussion

Occupant detection, classification and position is confirm based on the logic fusion of image and signal processing data. In image processing, the intelligent safety system is used two-data sets of images in the experiments to test the detection performance of human face and non-human object, which are distinct from the training sets. The first sets consist of 253 test images, whose have a wide variety of complex background in the various environment and scale changes for the object of interest along with some occlusion and variations in lighting. 25 human face image of interest is taken for a total of 253 test image. The second data sets contain 112 test images that have been collected from 7 non-human object of interest is taken. The systems undergo the bootstrapping cycle with ending up 4500 to 9500 zero samples, to evaluate the performance of true detection of the test images and the rate of false detection from the image of natural scenes that do not contain human face or non-human object.

Table 4 shows the performance of human face detection results of various methods on the test set 1 and compare with other systems in term of the number of detect faces, miss faces, false detection and computation time. The successful rate of the proposed method is 97.6 %, with 6 false alarms. It should be noted that the number of false alarms is quite small in compared to methods Yacoub et al. [11] and Fasel et al. [12] which is 347 false alarms. This may show the capability of the combination of two networks to highly separate human face from non-object examples. The higher performance of Rowley et al. 1998 [13] is likely due to the size of training data. We used a 7344 human face images and 8000 non-object examples, while Rowley et al. 1998 trained with 16000 face images and 9000 non-faces images. However, the technique is less efficient than our techniques in term of the false detection and response time. On the other hands, Yacoub et al. [11] shows a very fast time processing but have a drawback of higher false alarms.

Table 4. Detection Rate of Set 1 on Different Methods.

Method	Human Detect (%)	Miss Human (%)	No. False Detec	Process Time
FNN+CNN	97.63%	2.37%	6	2.3s
Rowley et al.	97.86%	2.14%	13	0.013M
Yacoub et al.	84.31%	15.69%	347	0.7s
Fasel et al.	96.8%	3.2%	278	3.1s

Similarly, Table 5 shows the summarized results of non-human object on the test set 2 and compare with other systems. We found that non-human object detection rate is 96.42%, which mean 108 out of 112 numbers of non-human objects are detected. The false detection rate is 3.58%, which is lower that Agarwal et al. [14] and others methods [15]. However, the average process time is almost same with others method providing additional calculation on CNN. Based on the results shown in Tables 4 and

5, we can concluded that both human face and non-human object detection system make acceptable tradeoffs between the number of false detection and detection rate.

Table 5. Detection rates of set 2 on different methods.

Method	N-Human Object Detect	Miss N-uman Object	No. False Det.	Proces s Time
FNN+CNN	96.42%	3.58%	4	2.9s
Agarwal et al.	94%	6%	30	3.6s
Mahmud & Hebert	82%	18%	187	4.0s
Viola & Jones	95%	5%	71	0.7s

Once the image processing part completed, the "ImageProcess" function provides 1 for human and 0 for non-human object. This 1 and 0 is fused with the weight sense of the sensor situated inside the vehicle seat to provide accurate occupant detection and classification for the integrated intelligent safety system. To illustrate the performance, some exemplary results obtained from the prototype system are demonstrated for instances such as, when the seat is occupied or empty. If occupied, the occupant is classified as an adult, child or non-human object. Usually, human in the seat provides its weight with positional variation. However, non-human object like grocery bag is static and provides its weight without positional variation. It also demonstrates that the position of the occupant can be determined for consideration of safety issues in airbag application or for measuring the comfort level.

Figure 4 shows the centroidal position of y_centroid vs. x_centroid of the vehicle front passenger seat of size 50x50 cm that indicates various position of the occupant such as standard, good and bad with marking blue, green and violet lines, respectively. Figure 4 (a) shows that occupant in the seat is good position with relax mode aligning to the back of the seat. Figure 4 (b) shows that the occupant is in bad position i.e. the occupant is aligned very much right to the seat. On the other hand, Figure 4 (c) indicates that occupant is in bad position and very close proximity to the airbag. In that case, our safety device will not provide decision for airbag deployment. Similarly, Figure 4 (d) also indicates that the occupant is in extreme left of the of the vehicle seat.

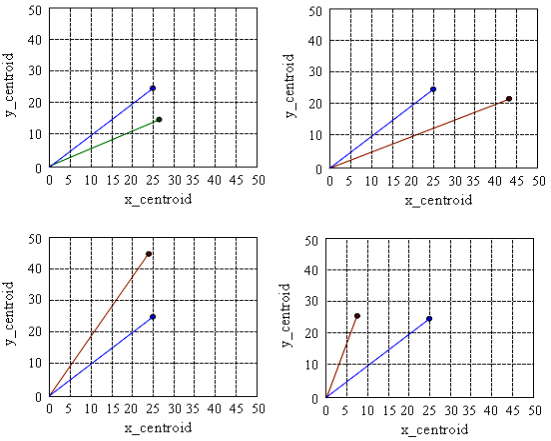


Fig. 4. Occupant centroidal position calculation.

Next, we have implemented the experiments of the frontal static crash using crash generating device and with the interface program. The experimental result of the crash reaction force being applied to generate crash is shown in Fig. 5. We have tried to obtain the reaction force during the repeated crash conduction at a time 51 sec to 80 sec. Figure 11 shows, during repeated crash, it gains a huge force of ~ 1000 N/m to ~ 5800 N/m and definitely the crash velocity immediately before the crash is greater than the 22.54 km/h. The reaction force depends on the crash velocity of the system. It is stated that as velocity increases the reaction force also increases, which in turn increases the crash severity. This is a situation that put the occupant at a higher risk.

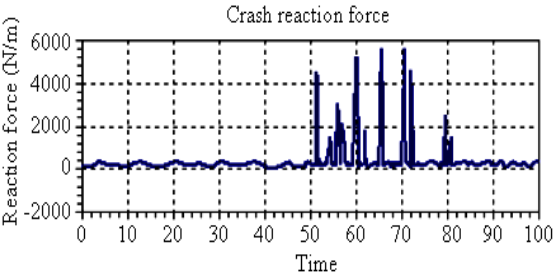


Fig. 5. Vehicle crash reaction forces.

For analysis of tire data, the 'Goodness of Fit' statistical analysis for the model is tested with all the variables. Figure 6 shows the 'Goodness of Fit' of the principle component analysis with polynomial fitting. The fitting result of parametric model of SSE is 0.1358 with 95% confidence bounds, which is close to 0 and indicates that the data fits well. The value of the multiple correlations R^2 coefficient is 0.8452, whilst the adjusted R^2 value is 0.8267. Both reveal about 85% and 83% match in the outcome which indicates a good fit. In addition, the RMSE value of 0.161, which is close to 0, also implies that the data fits well. In Fig. 6 (b), residuals of the polynomial fit appear to be randomly scattered around zero, which again indicates the model perfectly fits the data under the study. In short, we have statistically proven

that TPMS can play an important role to enhance tire safety, performance and maintain reliable operation in combine with vehicle intelligent safety system.

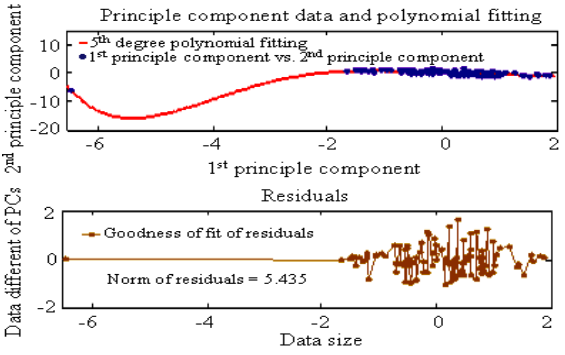


Fig. 6. Principle component analysis and goodness fit.

The Fig. 7 shows the display unit of experimental results and decision monitoring operation of the implemented integrated intelligent safety system towards ADDS and TPMS. The ADDS is involved on occupant detection, classification and position, vehicle crash detection and its severity analysis. The safety feature functions are activated by pressing start button.

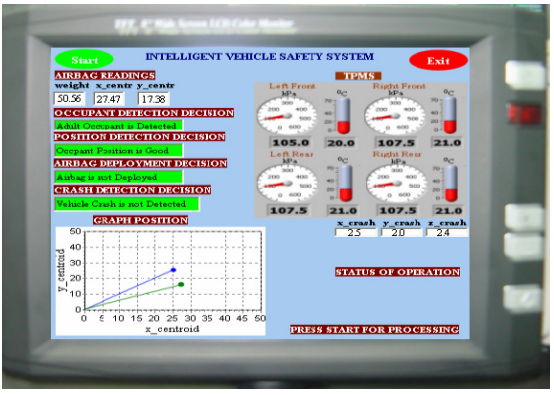


Fig. 7. Display of integrated intelligent safety system.

The display unit successfully achieved the decision application of safety features. Several safety issues related decision performances are fused using logic combination. In Fig. 7, its shows that when the system detect adult occupant and keeping good position, the airbag will not be deploy, provided there is no crash detected by the system. All the decisions in the display unit are shows as green mark. Accordingly, the icons of occupant total weight, centroidal position, axis wise crash data and position graph are displayed. For TPMS, the display unit also shows that all four tire acquiring the real-time temperature and pressure data, respectively. Therefore, the integrated prototype provides the optimum and their fused decision on various safety

issues that is very useful related to the safety issues for vehicle driving assistance system.

Similarly, when adult occupant is detected and keeping good position, airbag is deployed only whenever the vehicle crash is detected.

However, if the occupant position is bad i.e. occupant is very close to the airbag unit, the airbag would not deploy though the vehicle crash is detected. This is because of the huge reaction force would hit the occupant due to close proximity between occupant and airbag unit. This is a situation that put the occupant at a higher risk.

8 Conclusions

In this paper, the system interface of the integrated prototype implementation of vehicle intelligent safety system has been presented. Lab Window/CVI in C interface program is used for the real-time intelligent safety systems prototype implementation. The safety system such as TPMS, occupant detection, classification and position, vehicle crash detection and its severity analysis are integrated. The prototype is made by developing the algorithms and methodologies of the hardware platform and system interface program. The application of the embedded intelligent safety system has resulted in successful real-time working device, which will provides the validation of the performance diagnosis of the safety system. The contribution of the interface system of this prototype are performance characterization, problem determination and real-time work load data monitoring of a distributed safety issues and provides safety warning, whose are applicable in successfully operation.

Acknowledgements

The authors would like to thank the Malaysian Ministry of Science, Technology and Innovation (MOSTI) for funding this work through IRPA research grant 03-02-02-0017-SR0003/07-03.

References

1. D. S. Timothy, and M. M. Trivedi, “Real-time stereo-based vehicle occupant posture determination for intelligent airbag deployment” *Proc. of the IEEE International Conference on Intelligent Vehicles,* June, 2003, pp: 570-574.
2. M. Devy, A. Giralt, and A. Marin-Hernandez, “Detection and Classification of Passenger Seat Occupancy using Stereovision” *Proc. of the IEEE Intelligent Vehicle Symposium,* Dearborn (MI), USA, Oct. 3-5, 2000, pp: 714-719.
3. M. Shiraishi, H. Sumiya, and Y. Ysuchiya, “Crash Zones based on Driver’s Collision Avoidance Operation for ITS”. *Proc. of the IEEE Conf. on Intelligent Transportation System,* Singapore, 3-6 Sept. 2002, pp: 210-215.
4. H. Kong, Q. Sun, W. Bauson, S. Kiselewich, P. Ainslie and R. Hammoud “Disparity Based Image Segmentation For Occupant Classification” *Proc. of the IEEE Computer Society*

Conference on Computer Vision and Pattern Recognition Workshops(CVPRW'04), Vol. 1, 2004, pp. 126-134.

5. N.Shigeyuki, "Development of occupant classification system for advanced airbag requirements", *Mitsubishi Motors Technical Review*, 16: 61-64, 2004.
6. T. Marunaka, T. Kimura, M. Taguchi, T. Yoshikawa, H. kumamoto, and K. Kishida, "Study on the Crashworthiness of Rail Vehicles" *Proc. of the IEEE/ASME Railroad Conference*, 17-19 April, 2001, pp: 251-257.
7. A. Rovid, and G. Melegh, "Modelling of Road Vehicle Body Deformation Using EES Detection " *Proc. of the IEEE Conf. on Intelligent Signal Processing***,** Budapest, Hungary**,** 4-6 Sept. 2003, pp:149-154.
8. NTSB (National Transportation Safety Board), "Safety Recommendations H-97-19 through -21" http://www.ntsb.gov/recs/letters/1997/H97_19_21.pdf, Washington, D C 20594 (January 10, 2005).
9. Ronald K. J. 1989. Global 90 cars: electronics-aided. *IEEE Spectrum,* 26(12): 45-49.
10. Burgess, J. 2003. Application Note AN1951/D: Motorola Tire Pressure Monitor System Demo. *Freescale Semiconductor*, Inc.
11. S. Ben-Yacoub, B. Fasel and J. Luettin, "Fast Face Detection using MLP and FFT", *Proc. Second International Conf. On Audio and Video-based Biometric Person Authentication* (AVBPA '99), 1999.
12. B. Fasel, S. Ben-Yacoub and J. Luettin, "Fast Multi-Scale Face Detection", *IDIAP-Com 98-04*, 1998, pp. 1-87.
13. H. A. Rowley, S. Baluja, and T. Kanade, "Neural Network-Based Face Detection", *IEEE Transaction on Pattern Analysis and Machine Intellignce*, Vol. 20, No. 1, 1998, pp. 23-38.
14. S. Agarwal, A. Awan, and D. Roth, "Learning to Detect Objects in Images via a Sparse, Part-Based Representation", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 11, 2004, pp. 1475-1490.
15. S. Mahamud and M. Hebert, "The Optimal Distance Measure for Object Detection", *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR'03), Vol. 1, 2003, pp. 248-255.

Integration of Mobile RFID and Inertial Measurement for Indoor Tracking of Forklifts Moving Containers

M. Fodor[1], O. Gusikhin[1], E. Tseng[1] and W. Wang[2]

[1] Ford Motor Company
Research and Innovation Center
20000 Rotunda Drive, Dearborn, MI 48121, U.S.A.
[2] Ford Motor Company
Walton Hills Stamping Plant
7845 Northfield Road, Bedford, OH 44126, U.S.A.

Abstract. Described are the motivation and a method for combining readings of RFID tags placed in static locations in a plant environment using a mobile reader attached to a forklift and inertial measurements of the motion of the forklift to track its position and ultimately the positions of containers it moves within the plant. Strengths and limitations of RFID-based and inertial-based methods are presented along with an algorithm for postprocessing and combining the readings of each. The accuracy of the resulting position estimates are shown to be on the same order as the accuracy of RFID reader range variation.

1 Introduction

Desire for real-time visibility of inventory and assets across the supply chain is driving fast adoption of advanced information technology for location tracking. Location information eliminates the need for non-value added inventory search activities and creates a foundation to further optimize the operational efficiency of business units [1]. While the integration of RFID, GPS, and wireless communication is already common for the logistics industry to track products in routes between supply chain sites [2], tracking within indoor environments such as plants and warehouses is still a challenging problem.

Advancements in RFID technology have facilitated locating tagged assets indoors [3]. In particular, the field of Real Time Locating Systems (RTLS) is rapidly growing, primarily employing active RFID. Existing RTLS solutions differ in operating frequency, methods, granularity, accuracy, and the resulting cost of infrastructure and operation. Despite significant development, RTLS is presently prohibitive for commodity item tracking due to the substantial cost of active RFID and is typically employed only for tracking personnel and expensive resources.

Tracking commodity inventory or containers is usually accomplished using passive RFID and fixed readers operating at "choke points", providing zoning location of a given inventory item. For a large area, the infrastructure cost can be very high, and often fixed readers are installed only at shipping and receiving doors, providing inventory information but not location. The recent introduction of forklift-

mounted mobile RFID readers [4] addresses the problem of limited visibility into inventory locations associated with choke points. In this case the locations of the inventory items can be recorded from the location of the RFID reader at the point of unloading. Provided that these items will be moved using only vehicles equipped with the reader and the location of the reader is known, this method can offer relatively reliable location records. A number of different approaches have been proposed for localization of mobile readers. In addition to the RFID-based RTLS mentioned above [5, 6], other technologies include Wi-Fi, Ultrasonic, and Infrared. Often these network-centric approaches are hampered by the need for a substantial initial investment in infrastructure to ensure the required coverage and accuracy. An alternative approach relies on sensors and instrumentation installed on the delivery vehicle to track its location.

There is a substantial body of knowledge related to vehicle-centric localization methods developed within the field of robotics [7] and successfully applied in industrial settings [8]. The most common approaches are dead reckoning and the use of landmarks. Dead reckoning is a method of finding the relative position of a mobile device from a previous known position using inertial measurements or odometry. Challenges with this method include the need to know the original position and the accumulation of errors requiring continuous resetting of position using other sensors. Landmark-based localization determines the absolute position of the device through the recognition of predetermined distinct natural or artificial features of the environment. Artificial landmarks are location reference markers attached to walls, ceiling, or floor that can be easily recognized by vehicle-mounted instrumentation and can be relatively inexpensive to install. Examples include special visual patterns [9], infrared light-emitting diodes [10], and RFID tags [11-13].

Although both dead reckoning and landmark-based methods are error-prone, fusion of the two can result in relatively reliable localization. Several localization methods have been proposed that deal with uncertainty of the measurement data and provide data fusion from different noisy sources such as dead reckoning and landmarks. The Kalman Filter is a widely used method to compensate for noise and is applicable to the localization problem [14]. Monte-Carlo or Particle Filter localization [15] and fuzzy logic [16] have also been considered. In this paper we present an algorithm for the fusion of RFID-based localization and inertial measurement to obtain an accurate location of a delivery vehicle (forklift). With the forklift already equipped with a mobile RFID reader, it is logical to consider the use of passive RFID labels to create static location references in the environment. Since the application considered does not require instantaneous knowledge of an asset's location, the data from both inertial measurements and RFID are fused using post-processing.

This paper describes the automotive part stamping environment and the need for container tracking. Passive RFID and inertial-based localization are then presented with strengths and weaknesses of each individually. The synergistic fusion of the two methods is shown to eliminate the weaknesses of each and is further improved by post-processing. The data fusion algorithm is then described, followed by conclusions.

2 Background

The automotive stamping plant is a first tier supplier providing major components for the vehicle body including doors, fenders, roofs, etc. The stamping plant supplies parts to assembly plants and service facilities using truck or rail transportation. In general, the stamping process consists of blanking, die press, and assembly/welding operations. At the end of the press or assembly operations the parts are placed into metal racks that are used to store and transfer them between stamping operations and customers. Containers hold 8 to several hundred parts each and are unique to each part. Typically, the size of the racks range between 4 to 12 feet in length, 3 to 7 feet in width, 4 to 8 feet in height, and weigh up to 5000 lb when loaded. There are over 20,000 racks in any given plant, and for each given part type there is a limited rack fleet. It is important to closely monitor the flow of racks within the plant and between the plant and the customer's site. If the empty racks are not received back from the customer on time or are held at the repair area, there may be an insufficient number of racks to support production, resulting in non-optimal production batch sizes or hampering the ability of the plant to satisfy customer demand. Each rack is tagged with a passive RFID label, and fixed readers at shipping doors monitor rack flow between the plant and its customers. However, more granular tracking using fixed readers would require a substantial investment in infrastructure as a typical stamping facility is over 1,000,000 square feet.

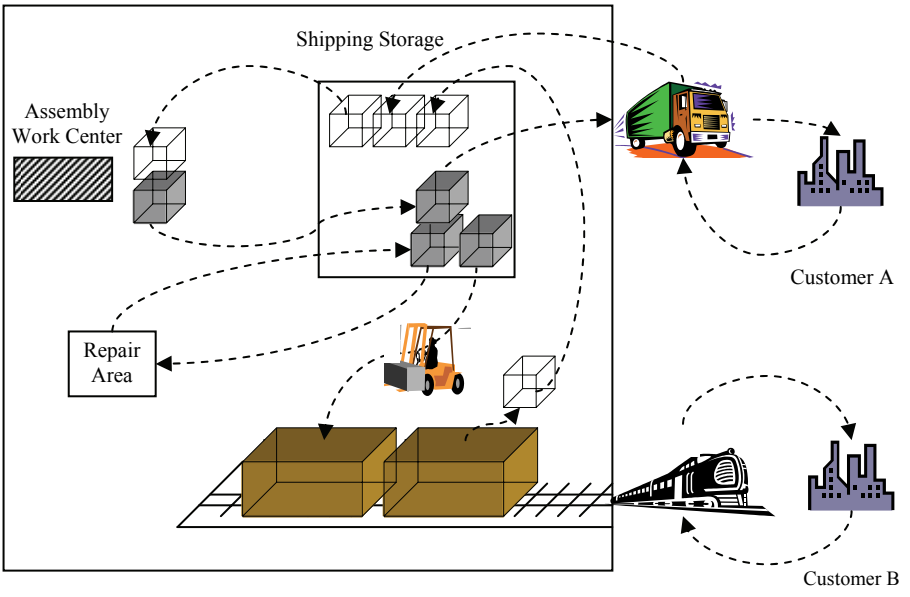


Fig. 1. Schematic of the flow of racks associated with the given work center.

The racks are handled by forklifts which move between the end of manufacturing lines and storage areas, to and from trucks and rail cars, or to and from a repair area. Figure 1 illustrates the rack flow within a typical stamping facility and associated with a given set of parts produced by an assembly-welding operation (work center). The

forklift transfers the racks with parts from the work center to the dedicated shipping storage area and brings empty racks back to the work center. These parts can then be loaded onto trucks or rail cars. If there are quality concerns regarding the parts stored in the shipping area, the forklift may move these parts into a repair area and then back. A typical stamping facility employs 40 forklifts, and each forklift is equipped with a wireless terminal that can exchange data with a back office computer.

The many benefits of knowing the specific location and status of racks (full, empty, or in repair) include saving time for material handling personnel locating inventory and reducing downtime caused by unavailability of racks. In addition, when parts are quarantined due to quality concerns, location information can reduce the number of racks that are pulled from inventory. Location and status information can also facilitate a first-in-first-out (FIFO) inventory control system, improve inventory turnover rate, reduce the potential for obsolete parts, and improve material flow and space utilization.

3 Inertial Tracking

Inertial measurement of motion involves the use of linear accelerometers and/or rotational rate sensors whose signals are mathematically integrated to produce speed and position estimates. These sensors are deployed in a wide array of applications including air, space, and ground vehicles as well as various consumer electronic systems. In each application, the orientation or dynamic motion state of the vehicle or device is of interest. Automotive applications include anti-lock braking, traction control, yaw and roll stability controls. Typical sensor sets for ground vehicles measure longitudinal and lateral vehicle accelerations and yaw rate, with roll rate seeing wider use now in roll mitigation systems. In automotive applications, vehicle wheel speed sensors and GPS may also be added.

A general-purpose method for tracking a system in two-dimensional space involves combining longitudinal and lateral accelerations and yaw rate. Yaw rate $\dot{\psi}$ (see Figure 2 for signal and axis definitions) is integrated to provide yaw angle or heading:

$$\psi(t) = \psi_0 + \int_0^t \dot{\psi}\, dt\,. \tag{1}$$

Vehicle-fixed longitudinal (x-axis) and lateral (y-axis) accelerations, a_x and a_y respectively, are then combined with heading and yaw rate to give velocities in the fixed frame of reference (with respect to inertial ground):

$$V_x(t) = V_{x,0} + \int_0^t \left[a_x \cos(\psi) - a_y \sin(\psi) - \dot{\psi} V_y\right] dt\,, \tag{2}$$

$$V_y(t) = V_{y,0} + \int_0^t \left[a_y \cos(\psi) + a_x \sin(\psi) + \dot{\psi} V_x\right] dt\,. \tag{3}$$

Finally, these inertial velocities are integrated to give positions in the inertial frame:

$$X(t) = X_0 + \int_0^t V_x \, dt \,, \tag{4}$$

$$Y(t) = Y_0 + \int_0^t V_y \, dt \,. \tag{5}$$

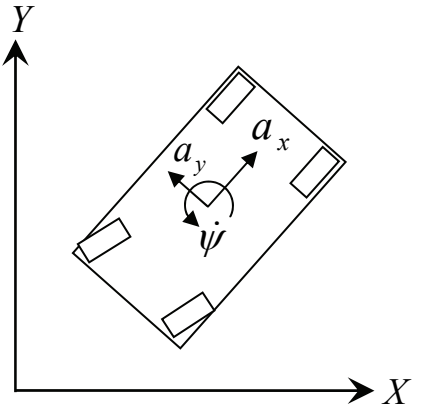


Fig. 2. IMU signals and axes.

Because the sensor measurements are prone to offset and noise errors, this double-integrating process results in error accumulation, limiting the time and distance over which this process provides acceptable results. These limitations can be mitigated to some degree by including kinematic constraints on the integration process which are imposed by the vehicle itself. For instance, the forward speed, yaw rate, and lateral acceleration of a wheeled vehicle are coupled directly as long as the vehicle moves without sliding its wheels (a good assumption for heavy, factory floor forklifts). Additionally, vehicle states such as speeds and rotational rates are bounded by the vehicle operating envelope. The inertial tracking method benefits from a continuous stream of data from the sensors, resulting in uninterrupted, fine-grained position information. However, to anchor the path calculation in absolute space, the position, heading, and velocity of the vehicle at the initiation of tracking (X_0, Y_0, $V_{x,0}$, $V_{y,0}$, and ψ_0) must be known by some other means. For this reason, in addition to the drift problem, inertial tracking alone is not completely suitable for forklift tracking.

4 RFID Grid Tracking

Unlike inertial tracking that drifts and has no inherent connection to absolute space, RFID tags used as a tracking system can be physically connected to the operating environment itself. RFID as a static positioning system requires the use of both tags as well as a reading device that receives their identification information. The tags are either attached at various fixed and known locations in the plant environment and detected with a reader attached to the forklift, or they are deployed on the forklift and

read with a plurality of stationary readers. The former approach is advantageous as it also facilitates the detection of the tagged and tracked inventory (racks). If the location tags are deployed in a fine enough mesh within the plant environment, forklift tracking can be accomplished using this method alone. However, despite their low cost, passive RFID tags must still be installed, cataloged, and maintained and are subject to damage in the hostile plant environment where suitable safe installation locations may be few and far between. For example, a typical stamping plant may have support columns, the only suitable installation location for tags, separated by 40 feet or more.

There are other inaccuracies as well. Because the tags can be read at a distance from a range of orientations, the exact location of the reading device attached to the tracked vehicle is not known when communication is acquired with a tag. Therefore, the static RFID tag location signal has some position error since the passing reading device can only be assumed to be located within some expected range (10 to 15 feet) of the energized tag. Additionally, there may be time when the tracked vehicle is not in range of any tag or where a tag is damaged and inoperative. Thus the RFID tracking method, while attached to inertial space, is nevertheless somewhat inaccurate, discrete in nature, and intermittent.

5 IMU and RFID Fusion

Combining the inertial (IMU) and static (RFID) tag data can reduce or eliminate the limitations associated with each individually. The inertial measurements provide continuous data that is potentially accurate enough to reasonably "connect the dots" between the sparse static tag data. Using the absolute position data from the RFID tags, offset errors in the IMU readings can be estimated and removed, resulting in improved positioning, and the IMU data integration can be initialized with static RFID location information to connect it to absolute space.

In the automotive stamping plant setting, the instantaneous location of a tracked vehicle is less important than an accurate estimate of the vehicle's path during a finished delivery as each event takes a few minutes or less. Therefore, the method proposed here assumes that inertial sensor data and static tag data will be collected and stored during an event. When the event ends, the data will be post-processed to determine the path of the delivery vehicle (especially its end points). Using data from the entire event provides a richer data set from which to calculate path, whereas attempting a continuous, immediate position estimate during the delivery event limits the calculations to data that has occurred in the past only and provides no real benefit as the post-processed data is timely enough (within a few seconds of the event ending) for the stamping environment.

6 Postprocessing using Best-Fit Optimization

The proposed computational method relies upon a recursive solution that makes repeated guesses of the initial vehicle position and IMU sensor offsets to attempt to

minimize the error between the resulting IMU-based path estimate and the known locations of the static RFID position readings taken over that path. As the solution converges, RFID reader range information is employed to attempt to predict the RFID signal acquisition and loss locations to further improve solution fidelity. The full method is consists of the following 6 steps.

Step 1. Inertial (IMU) and static tag (RFID) data is collected during an inventory delivery event. Inertial data is collected continuously at a regular rate, typically 10 to 100 Hz. Static tag data consists of the time and ID number of each tag as communication with the tag is acquired and then lost.

Step 2. Upon completion of the delivery event (determined by observing acquisition and loss of inventory tag readings), the vehicle path is reconstructed solely from the inertial data by making vehicle initial state assumptions (X_0, Y_0, $V_{x,0}$, $V_{y,0}$, and ψ_0). Typically, these initial conditions are chosen based on information from the end of the previous delivery event.

Step 3. Using a best guess position for each RFID tag reading, one that represents the most likely average vehicle position while it is in communication range with the location tag, a path error calculation is made using the path constructed in Step 2:

$$E = \frac{1}{n}\sum_{1}^{n}\sqrt{\left[X(t_n) - X_{RFID,n}\right]^2 + \left[Y(t_n) - Y_{RFID,n}\right]^2} \quad . \tag{6}$$

Here n is an index indicating the acquisition and loss events for the RFID tag readings, and $X(t_n)$ and $Y(t_n)$ give the position of the IMU integration calculation (Equations 1-5) at these RFID tag detection times. The best guess positions, $X_{RFID,n}$ and $Y_{RFID,n}$, are unique for each tag and are based on the position of the tag, the most likely vehicle path followed when in range of the tag, and the reader's expected communication range. These values can be cataloged during the tag's installation or constructed from data collected during system operation.

Step 4. The assumed vehicle initial state (first used in Step 2) is perturbed, and Steps 2 and 3 are repeated until these parameters converge to minimize the error calculated in Step 3. This iterative process can be conducted using any of a number of optimization algorithms such as Matlab's *fminsearch* function. The result of this step is the best match of the inertial sensor-based position to the best guess RFID tag positions used in Step 3.

Step 5. The result of Step 4 is used to calculate the most likely RFID tag acquisition and loss locations using detection range assumptions. At each time of acquisition or loss, the vehicle position (from Step 4) and tag location data are used to find the intersection of the estimated vehicle path with the locus of expected tag detection range points. This locus can be assumed to be circular, or more complex shapes can be used based on more detailed tag and reader information. These new

acquisition/loss points establish a more likely vehicle path for further optimization of the vehicle initial conditions (X_0, Y_0, $V_{x,0}$, $V_{y,0}$, and ψ_0).

Step 6. Steps 3 and 4 are repeated using the new acquisition/loss positions produced in Step 5. These positions are updated with each iteration of the inertial data path optimization, and this step is repeated until the solution converges and each subsequent iteration produces a path prediction that is negligibly different from the iteration before.

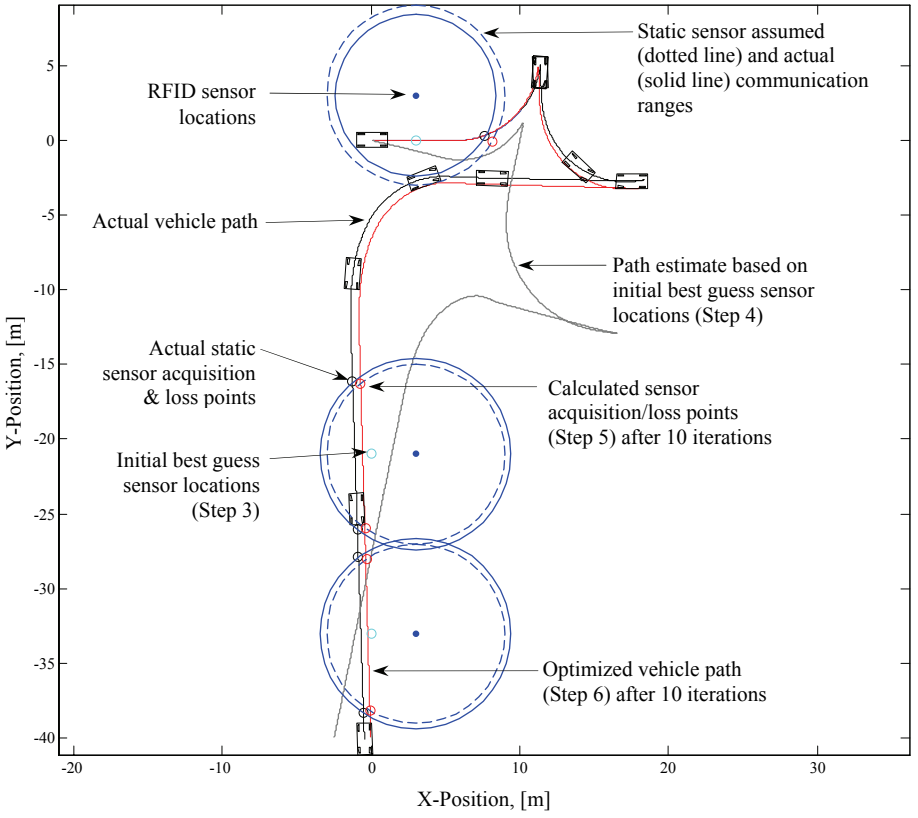


Fig. 3. Forklift path reconstruction showing actual forklift position evenly spaced in time, RFID antenna locations and ranges, initial path estimate, and final path estimate after ten iterations.

Figure 3 illustrates several of the aspects of the method described above. The actual path of the forklift is shown for a delivery event with forklift orientation superimposed at regular time intervals. Large solid circles surround the RFID location tags and depict their actual detection range, while the large dashed circles are the assumed reader ranges used in Step 5. The smaller circles located within the tag

detection range are the initial best guess detection locations used in Step 3, and the half-tone line is the path resulting from the optimization using these points (result of Step 4). Shown also is the result of the 10th iteration of Step 6 using updated acquisition and loss points calculated in Step 5. Note the convergence of the optimized path toward the actual path from the Step 4 result to the Step 6 result.

The accuracy of the path reconstruction depends mostly on the accuracy of the assumed reader detection range as compared to the actual range. Note from the figure that the path error (the distance between the actual and final estimated path) is roughly the same size as the reader range variability. For a typical forklift-mounted RFID reader, this is 5 – 10 feet. Solution accuracy can be further improved by including inertial sensor offset errors that are optimized with the initial conditions.

7 Conclusions

Presented here is an effective method of indoor localization of forklift vehicles equipped with mobile RFID readers. The method is based on the fusion of inertial measurements with information from static RFID location tags. While each individual method alone is prone to substantial errors, fusion of the two can provide results of reasonable quality while minimizing the cost of implementation. The accuracy of the resulting location estimate depends on the spacing of the location tags throughout the plant as well as the variability of the RFID reader detection range. In practice it is possible to achieve an accuracy within 5-7 feet which is acceptable for the stamping plant problem.

The ultimate goal of this development is to determine and record the locations of the RFID tagged containers handled by a given forklift. Specifically, the described approach has been developed to track the rack locations for automotive stamping plants. While this paper has focused on the fusion of the inertial measurements with the static information from the RFID tag locations for the purpose of tracking the delivery vehicle, it should be noted that the full material tracking problem is more complex. In addition to location awareness, the pick-up and delivery event with its associated inventory must be identified by monitoring the stream of RFID tag readings seen by the mobile reader. This must be accomplished robustly despite delivery complexities such as moving stacks of racks where not all of the moved racks will be seen by the reader during the entirety of the delivery event.

References

1. Thiesse, F., Fleisch, E.: On the Value of Location Information to Lot Scheduling in Complex Manufacturing Processes, Int. Journal of Production Economics, Vol. 112, 2008, pp. 532-547.
2. Jones, E.C., Chung, C.A.: RFID in Logistics: A Practical Introduction, CRC Press, 2008.
3. Sanpechuda, T., Kovavisaruch, L.**:** A review of RFID localization: Applications and Techniques, Proceedings of 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology ECTI-CON 2008, pp. 769 – 772.

4. Jungk, A., Heiserich, G., Overmeyer, L.: Forklift Trucks as Mobile Radio Frequency Identification Antenna Gates in Material Flow, Proceeding of Conference on Intelligent Transportation Systems, 2007, pp. 940 – 943.
5. Michel, J. C. F., Millner, H., Vossiek, M.: A Novell Wireless Forklift Positioning System for Indoor and Outdoor Use, 5th Workshop on Positioning, Navigation and Communication 2008 (WPNC'08), Hannover, Germany, March 2008, pp. 219-227.
6. Röhrig, C., Spieker, S.: Tracking of Transport Vehicles for Warehouse Management using a Wireless Sensor Network, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, September 22-26, 2008, pp. 3260-3265.
7. Borenstein, J., Everett, B., Feng, L.: Navigating Mobile Robots: Systems and Techniques, A. K. Peters, Ltd., Wellesley, MA, 1996.
8. Hu H., Gu D.: Landmark-based Navigation of Industrial Mobile Robots, International Journal of Industry Robot, Vol. 27, No. 6, 2000, pp. 458 – 467.
9. Yoon, K.-J., Kweon, I. S., Lee, C.-H., Oh, J.-K., Yeo, I.-T.: Landmark Design and Real-Time Landmark Tracking Using Color Histogram for Mobile Robot Localization, International Symposium on Robotics (ISR), 2001, pp. 1083-1088.
10. Lee, S., Song, J.-B.: Use of coded infrared light as artificial landmarks for mobile robot localization, Proceeding of International Conference on Intelligent Robots and Systems, 2007, pp. 1731-1736.
11. Hahnel, D., Burgard, W., Fox, D., Fishkin, K., Philipose, M.: Mapping and Localization with RFID Technology, Proceedings of ICRA 2004, pp. 1015-1020.
12. Schneegans, S., Vorst, P., Zell, A.: Using RFID Snapshots for Mobile Robot Self-Localization, Proceedings of the 3rd European Conference on Mobile Robots (ECMR 2007), Freiburg, Germany, 2007, pp. 241-246.
13. Yang, P., Wenyan, W., Moniri, M., Chibelushi, C.C.: SLAM Algorithm for 2D Object Trajectory Tracking based on RFID Passive Tags, Proceeding of 2008 IEEE International Conference on RFID, 2008, pp.165 – 172.
14. Santana, A.M., Sousa, A. S., Britto, R.S., Alsina, P.J., Medeiros, A.D.: Localization of a Mobile Robot Based in Odometry and Natural Landmarks Using Extended Kalman Filter, Proceedings of 5th International Conference on Informatics in Control, Automation and Robotics, 2008, pp. 187-193.
15. Thrun, S., Fox, D., Burgard, W., Dellaert, F.: Robust Monte Carlo Localization for Mobile Robots, Artificial Intelligence, Vol. 128, 2001, pp. 99-141.
16. Demirli, K., Molhim, M.: Fuzzy dynamic localization for mobile robots, Fuzzy Sets and Systems, Vol. 144, 2004, pp. 251–283.

POSTERS

Road-Following and Traffic Analysis using High-Resolution Remote Sensing Imagery

Seyed Mostafa Mousavi Kahaki [1], Mahmood Fathy [2] and Mohsen Ganj [3]

[1] Computer Eng. Department
Islamic Azad University Dezful Branch, Iran
mostafa.mosavi@yahoo.com
[2] Associate Professor Computer Eng. Department
Iran University of Science & Technology, Iran
mahfathy@iust.ac.ir
[3] Computer Eng. Department
Islamic Azad University Shoshtar Branch, Iran
mohsenganj@yahoo.com

Abstract. As vehicle population increases, ITS (Intelligent Transportation Systems) becomes more significant and mandatory in today's overpopulated world. Vital problems in transportation such as mobility and safety of transportation are considered more, especially in metropolitans and road ways. Road traffic monitoring aims at the acquisition and analysis of traffic figures, such as presence and numbers of vehicles, and automatic driver warning systems are developed mainly for localization and safety purposes. In this paper we propose a strategy for road following from aerial images. Real time extraction and localization of a road from an aerial image is an emerging research area that can be applied to vision-based traffic controlling and navigation of unmanned air vehicles. In order to deal with the high complexity of this type of images, we integrate detailed knowledge about roads using explicitly formulated scale-dependent models. The intensity images are used for the extraction of road from aerial images. Threshold techniques, Hough transform and learning algorithm are used for the road extraction and car detection. The results show that the proposed approach has a good detection performance.

1 Introduction

Today satellite and airborne remote sensing systems can provide large volumes of data that are invaluable in monitoring Earth resources and the effects of human activities. Traffic control is an emerging research topic due to rapidly increasing interest in their use. Currently, traffic control is a difficult and time consuming task that need too several human operators. Traffic controlling on satellite image can save time and costs. The urban traffic control process is accomplished through cameras which are installed in highways, in current technology. Recently, with considering aerial imageries, existence of an Intelligence road Extraction and car detection system which be able to control the road traffic would have more remarkable performance, Investigations about road Extraction and car detection in aerial imageries Involved in informa-

tion and data related to GIS and this maintained data needs to become up to date in every certain period of time. Road Extraction and Car detection in aerial Imageries is a modern Controversy issue in computer vision science that has also some influences on many other projects and operations. This paper proposed an integrated approach for automatic road extraction from remotely sensed imagery by combining digital image processing, remote sensing and Geographic Information System (GIS) technologies, Since the launch of new optical satellite systems like IKONOS, Quick Bird and Geoeye-1, this kind of imagery is available with 0.4 - 1.0 meter resolution. Vehicles can be observed clearly on these high resolution satellite images. Results show that the proposed algorithm has a good detection performance. Some vehicle detection methods have been studied using aerial imagery[1][7][9]. The Major factors that effect essentially on our subject are: the number of different objects in Imagery, amount of relationship between them and some properties that distinguish them from other objects. Example system for traffic monitoring has shown in Figure 1.

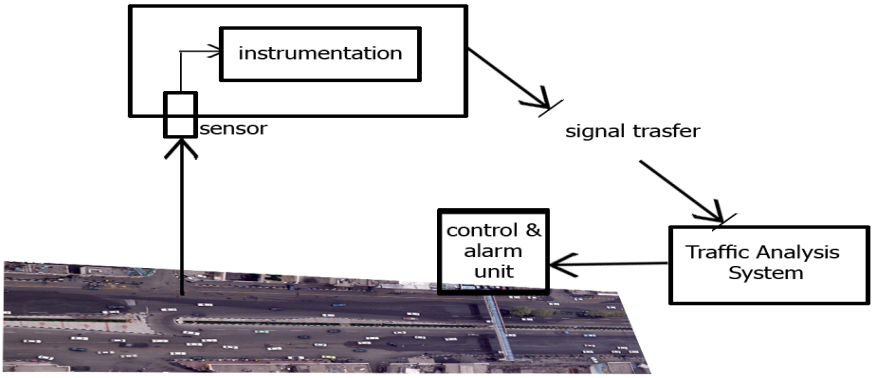


Fig. 1. Example system for traffic monitoring (Azadi Avenue, Tehran-IRAN).

Before explaining any detail, we scan other parts of this article:
Sect.2: Important Issues in road Extraction and car detection process.
Sect.3: Main idea and road and car, essential items and their characteristics.
Sect.4: An execution on aerial Imageries, advantages and disadvantages.
Sect.5: A proposal about works for future on this topic.

2 Related Works

Vehicle detection and road extraction has been receiving attention in the computer vision. A number of conventional express way incident detection algorithms have been developed in the recent years[11][12]. Techniques based on morphology and neural network for vehicle detection and road extraction had developed in machine vision, but only a few researches have investigated the detection of traffic sensing on aerial images[13][14]. In this paper we used high resolution images from Geoeye-1 satellite. This imagery are taken full color and in equal interval from the ground. Geoeye-1 is equipped with the most sophisticated technology ever used in a commer-

cial satellite system. It offers unprecedented spatial resolution by simultaneously acquiring 0.41-meter panchromatic and 1.65-meter multispectral imagery. The used method is presented on figure 2 as a flowchart:

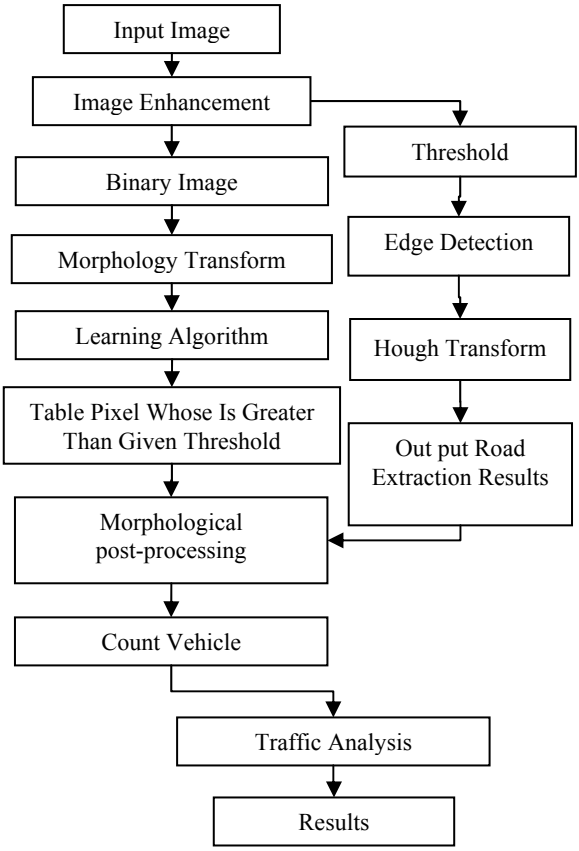


Fig. 2. Flowchart of the traffic analysis algorithm.

3 Vehicle Detection and Road Extraction Approach

First, we must extract the roads and distinguish coordinates of these roads's district; the car detection's operation is accomplished. In addition, we need feature extraction of car and road because other detection of road's district, only those objects can studied that are in the road district. This is really useful to know that roads are presented as a direct district with a different color in the aerial imageries. Therefore, linear feature can be an appropriate feature in detection at road's district, as you can see in Figure 3(a). Another feature which roads possess, Is the lines that exist in white color in all the roods and they are considered as a proper feature in detection operation. These lines are available continuously along side the road and discontinuously in the middle of the road. The other feature that used in this method is the color of the road,

which completely distinguishes between road and other parts. With threshold operation, we can separate road and background from each other and preparing imagery for other operations. This color threshold can be approached with mean of colors at the roods in several aerial imageries that are available in dataset. After threshold, edge detection operation can be run better. We had used the method for edge detection. This method has higher performance rather than the others. You will see the result in figure 3(d). In digital imageries, in those points that there is edge on them, there are differences of color too. Therefore, sharpen operation leads to increase the difference at color in edges and it can enhance the image classification. So, next operation like edge detection can be executed with higher precision.

In recent years the Hough transform and the related Radon transform have received much attention. These two transforms are able to transform two dimensional images with lines into a domain of possible line parameters, where each line in the image will give a peak positioned at the corresponding line parameters. This has lead to many line detection applications within image processing, computer vision. In the last step, we use Hough transform and Radon transform for distinguished road and direction detection for road following in aerial images as shown in figure 3 (d). We extract the edge lines angle to find direction of road. Several definitions of the Radon transform exists, but they are related, and a very popular form expresses lines in the form $\rho = x \cos \theta + y \sin \theta$, where θ is the angle and ρ the smallest distance to the origin of the coordinate system. As shown in the two following definitions (which are identical), the Radon transform for a set of parameters (ρ, θ) is the line integral through the image g(x, y), where the line is positioned corresponding to the value of (ρ, θ). The δ ( ) is the delta function which is infinite for argument 0 and zero for all other arguments (it integrates to one).

$$g(p,\theta) = \int_{-\infty-\infty}^{+\infty+\infty} g(x,y)\,\delta(p - x\cos\theta - y\sin\theta)\,dx\,dy \tag{1}$$

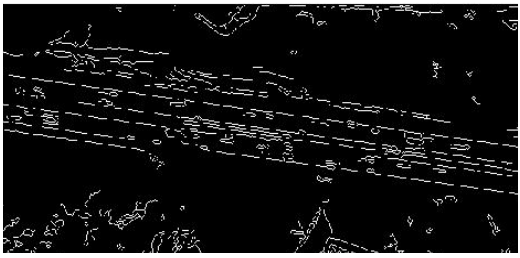
Or the identical expression

$$g(p,\theta) = \int_{-\infty}^{+\infty} g(p\cos\theta - s\sin\theta,\ p\sin\theta + s\cos\theta)\,ds \tag{2}$$

In his Ph.D. thesis [10], Peter Toft investigated the relationship of Radon transform with the Hough transform, and it is shown that the Radon transform and the Hough transform are related but not the same. The Radon transform of a function f(x,y) is defined as the integral along a straight line defined by its distance *P* from the origin and its angle of inclination θ , a definition very close to that of the Hough transform and requires a lot of processing power in order to be able to do its work in a reasonably finite time. Now a day high processing power is not a problem. Here we are considering all the line has same skew angle and the range of angle is -45° to 45°. Here Radon transform will detect the angle from the upper envelope. If the skewed angle is more than 45° or less then -45° the upper envelope may contain 2 lines in different directions. An example is shown in Figure 3(d) having 20 degree of skewed angle.

(a)

(b)



(c)

(d)

Fig. 3. Road Extraction (a) An original image (b) The Threshold pre-processing result (c) The Edge detection pre-processing result (d) The Road Extraction result and direction.

After road extraction and finding orientation of road, we need to detect the cars in the road. This trend is moved simply because only those objects can be handled that are inside the road. One of the important textures of cars, which helps us to detect, is car model. We can detect a car with using its model from dataset examples by using neural network classification.

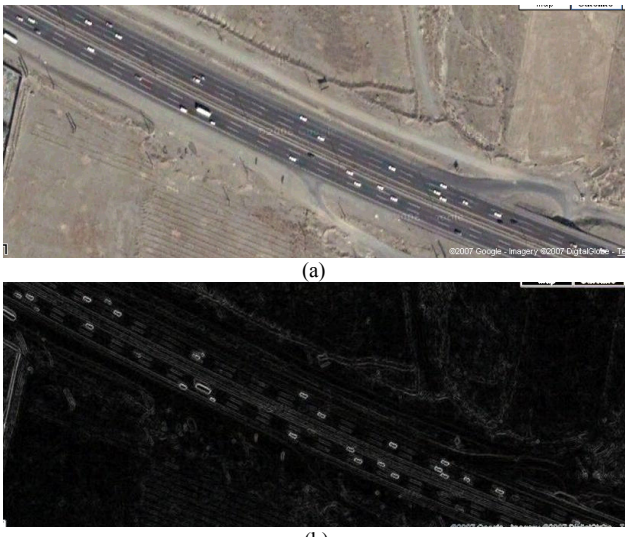


Fig. 4. (a) An original image (b) The morphology pre-processing result.

The dataset examples for car detection has extracted as shown in figure 5.

Fig. 5. Dataset examples.

After road extraction, cars should be counted for traffic analysis. Cars always in aerial imageries are appearing in rectangular shape as you can see in figure 4(b). Morphological Gradient use to enhance vehicle features. It is defined by Gradient:

$$G(f) = (f \oplus g) - (f \Theta g)$$

Where g is a structuring element, f is a gray scale source image, $f \oplus g$ means dilation operation, and $f \Theta$ g means erosion operation.

Figure 4 shows an original image of Tehran-Karaj highway and its morphology Gradient processing result. This process can use to discriminate vehicle targets and non-vehicle targets. Figure 5 shows some dataset examples for neural network classification. One noticeable feature in aerial imageries is the color of the cars that has a significant difference with other objects. This phenomenon can help us to detect the cars by adaptive threshold processing and the results of threshold simulation indicate that this method has high performance. After classification with neural network method the cars should be counted for traffic analysis. After counting cars, we can determine the amount of traffic in roads. Since all the aerial imageries are take from a fixed interval, the amount of road traffic with considering the cars number in one certain district is determined by adaptive threshold.

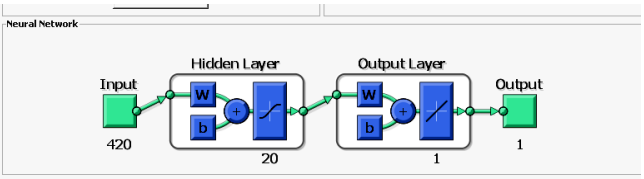


Fig. 6. Combining input information as neural network's input parameters.

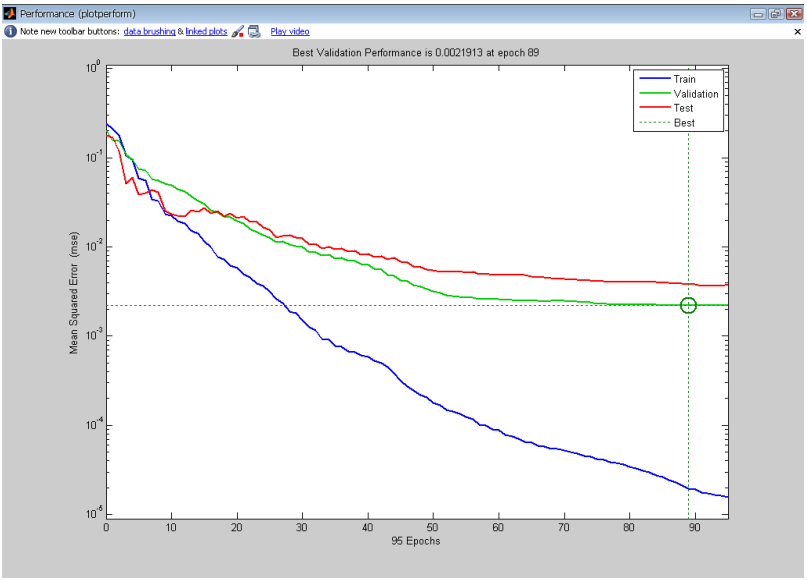


Fig. 7. Mean Square Error for Learning System.

This scheme, of course, has some disadvantages: its performance is about 90% and it can not detect the cars out side the road. This scheme can accelerate to transfer the information to drivers who are intended to cross a road and it also can help the police to traffic control. You can see the performance table of car detection in road at table1.

Table 1. Vehicle detection results.

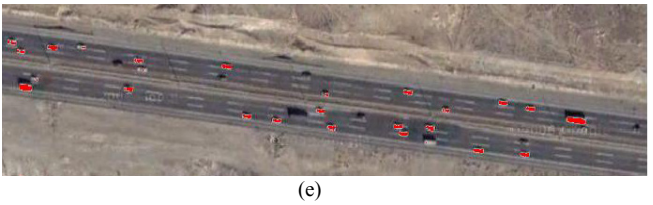
Site	Number of cars	Number of detected cars	Number of undetected cars	Performance
Road1	8	7	1	87.5%
Road2	52	47	5	90.3%
Road3	31	24	7	77.41%
Road4	30	27	3	90.3%
Road5	19	17	2	89.5%
Road6	55	44	11	80%

(a)

(b)

(c)

(d)



(e)

Fig. 8. Vehicle detection and Road Extraction results. (a)(d) The sample image of road segments. (b)The road extraction result, (c)(e)The Vehicle detection result, for image shown in (b), where lines represent extracted roads and for image shown in (c)(e), where dots represent detected vehicles.

After car detection and count cars, we can extract number of cars on a certain part of road and can analysis the traffic on it as shown in figure 9.

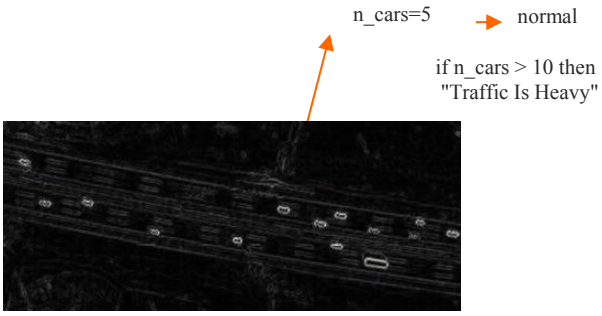


Fig. 9. Counting cars and simple traffic analysis

4 Conclusions

In this paper, we focus on the issue of vehicle detection and road extraction from high resolution satellite imagery for traffic analysis. Further work could include more training samples for neural network classification, and fusing more information like edge shapes to improve the correct detection rate. And also can detect an accident from high resolution satellite images. Imageries are taken by satellites and special planes at the moment. We can control the road's traffic via a traffic balloon. Traffic balloons are more flexibility to apply: There is possibility to pursue the roads via traffic balloon as auto pilot. Another capability of this intelligence system is to follow the deviant cars which have illegal speed.

References

1. Hong, Z. & Li, L.: Artificial Immune Approach for Vehicle Detection from High Resolution Space Imagery, IJCSNS International Journal of Computer Science and Network Security (2007) VOL.7 No.2.
2. Juozapavicius, A., Blake R., Kazimianec, M.: Image Processing in Road Traffic Analysis, Nonlinear Analysis: Modelling and Control, Nonlinear Analysis: Modelling and Control (2005) Vol. 10, No. 4, 315–332.
3. Higashikubo M., Hinenoya T., Takeuchi K.: Queue Length Measurement Using an Image Processing Sensor, In 3rd AnnualWorld Congress on ITS, Orland (2005).
4. Ho L. W., Kleeman L.: Real Time Object Tracking using Reflectional Symmetry and Motion, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2006).
5. Gonzalez R.C., Woods, R. E.: Digital Image Processing (2nd ed.), New Jersey: Prentice Hall (2002).
6. Gonzalez Rafael C. & Woods Richard E. & Eddins Steven L.: Digital Image Processing Using MATLAB (Hardcover), Prentice Hall, 1st edition.

7. Stefan, H.: Automatic Road Extraction in Urban Scenes-and- Beyond, Remote Sensing Technology, TU Munchen (2005).
8. Geoffrey, A., Hollinger: Design and Construction of an Indoor Robotic Blimp for Urban Search and Rescue Tasks (2005).
9. Zhu Z. & Lu H., Hu J., Uchimura K.: Car Detection Based on Multi-Cues Integration, IEEE 17th International Conference on Pattern Recognition (ICPR'04) (2004).
10. Peter Toft: The Radon Transform - Theory and Implementation, Ph.D. thesis. Department of Mathematical Modeling, Technical University of Denmark (1996).
11. Darbari M. , Sanjay M. and Abhay K. S.: Development of Effective Urban Road Traffic Management Using Workflow Techniques for Upcaming Metrocities like Lucknow (India), International Journal of Security and ITS Applications (2008) Vol. 2, No. 2, April.
12. Yong K. K.: Accident Detection System using Image Processing and MDR, IJCSNS International Journal of Computer Science and Network Security (2007) VOL.7 No.3.
13. Gerhardinger A., D. Ehrlich , M. Pesaresi: Vehicle Detection from Very High Resolution Satellite Imagery, IAPRS (2005) Vol. XXXVI, Part 3/W24 --- Vienna, Austria, August 29-30.
14. Hinz, Stefan: Detection and Counting of Cars in Aerial Images, Processding of The International Conference of Image Processing (2003) vol.3. Barcelona.

Vision Based Surveillance System using Low-Cost UAV

Kim Jonghun, Lee Daewoo, Cho Kyeumrae, Jo Seonyong
Kim Jungho and Han Dongin

Dept. of Aerospace Engineering, Pusan National University
Geumjung Gu, Busan, Korea
{Kjh1053,baenggi,krcho,whtjsdud123
kimsmap,sunnyban}@pusan.ac.kr

Abstract. This paper describes the development of a surveillance system in the laboratory for small unmanned aerial vehicles (UAV). This system is an important equipment of a mission-oriented UAV. For making a good performance on search, this can track images and take 3-D measurement of a target as well as acquire high quality images. Image tracking is carried out by the Kalman Filter. The position of the target in an image and the relationship among the coordinate systems of the UAV and the Camera and reference are used to solve the 3-D position of the target in real coordinates. This paper presents the hardware system as well as algorithm for the EOS, and then verifies the performance of the image tracking and real-time 3-D measurement of a target's position. Especially, to reduce the 3-D measurement error of the target, Linear Parameter Varying (LPV) is applied to the measurement system. The performances for their algorithms are presented in the figures in this paper.

1 Introduction

Recently, interest about UAV system has been increased. Basically, guidance and control were researched. After UAV system had ability to make a stable performance, some researchers are finding new algorithm to make a good result, like as neural network, fuzzy algorithm. Others make efforts to get a satisfied result using the exact measurement, UAV's attitude, position and velocity and target's position and velocity.

To get the accurate measurement about UAV and target, INS and GPS system are being studied. Also, machine vision system is being researched, recently. Vision system on UAV is necessary to acquire information of target and operate missions, such as surveillance and rescue at mountain or ocean. For civil system, to observe fire place at mountain is researched [1], and searching algorithm to track person or car at the urban is studied [2].

To measure 3D position of target using monocular vision, perspective matrix is used mostly [3]. This is easily to solve the position in image or real world with consider attitude between camera and target. However, algebra equations from perspec-

tive matrix are under determinant. To get the answer about the position, we need to know some points.

On the contrast, we can know distance between UAV and ground using GIS. If altitude of UAV and target in ground is known, perspective matrix is not necessary. We can acquire position through only relative equations about image and real world coordinate system. In this paper, nonlinear filter is employed to estimate 3-D position of target. Filter gains of this are derived by linear parametric varying system of error dynamics. This algorithm will be introduced and verified at chapter 4. Chapter 2 explains guidance/control, and hardware system on our UAV. Chapter 3 represents vision system.

2 UAV System

2.1 System Description

In this paper, the UAV system was used to study and design the algorithm that will allow image tracking and 3-D measurement of the position of the target in real space using a monocular image. Specification of UAV, used for experiment is PNUAV that we made ourselves, is represented in Table 1.

Table 1. Specification of PNUAV.

Wing Span	2,050mm
Wing Area	79.5 sq·dm
Empty Weight	3,300 g
Fuselage	1,633 mm
Class of Engine	19.96 CC

The flight control system consists of the main controller, GPS, Attitude Heading Reference System (AHRS), actuator, communication system, power supply and ground control system (GCS).

The AHRS offer the plane's attitude and position information to the sub-processor, that is, the main processor, which calculates algorithm to control the surface of plane directly via the actuator. Also, the states of plane can be observed at the ground station via the communication system, which can be used to transmit the control command to UAV from ground station when needed.

The GPS receiver that was used has comparative high performance and authoritativeness by altitude range in 18km and speed range in nearly 1850km/h that measuring is possible, produced by NovAtel Inc. Data output mode used NMEA, and used the GSA protocol which supplied position information and GGA protocol which included satellite information.

The RF data modem is used as communication device that receive and confirm all sensor data of UAV, when UAV is in flight. The antenna in the modem greatly affects the communication equipments. A directional antenna has a big gain value, but the UAV uses non-directional dipole antenna to considering the UAV has wide active area.

2.2 Guidance & Control

The UAV has various waypoints, where it can obtain images and drop the something (like a bomb), so the correct/accurate passing of the waypoint is a very important performance index of the operation of the UAV. Point navigation guidance is used in the flight test. The guidance logic is based on the difference between the line of sight angle, which is measured from the reference axis to the waypoint, and the UAV's heading angle, which is set as an error, that is made to be zero. In a straight level flight, the angle of the velocity vector and the LOS angle are used for lateral direction control. The longitudinal direction guidance uses proportional control that is based on the difference between the present altitude and the target altitude.

When only the P controller is used, the overshoot increases. On the other hand, if the PID controller is used, the steady state error will be reduced. However, I gain has less influence on flight performance and the computation load is added to the main processor. For this reason, the PD controller was used. It used the fluxion of angle, so it makes different to the weight of control. That is, the effect of fluxion is small when the flight to straight. However, if the LOS angle grows, the control value will increase, and then the UAV will show a fast response.

When the UAV passes a waypoint, without converging, the waypoint is in the minimum turning range. The UAV will fly, turning many times. Therefore, in this case, the UAV must escape the minimum range to follow the former control command. Fig. 1, it shows the geometric relations of UAV's flight range and the waypoint.

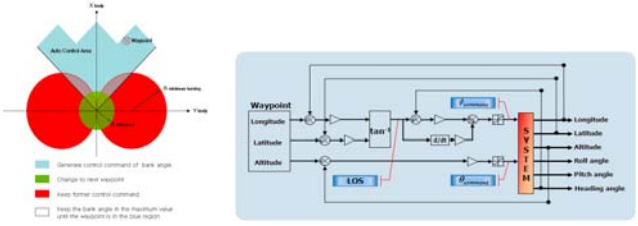


Fig. 1. control range at body axis (left), design PD controller (right).

3 Vision System

The vision system of PNUAV consists of a transmission system, gimbal controller, and image acquisition system. The gimbal controller provides the control input for pan and tilt movements. Camera zoom and focus are also controlled by the gimbal controller.

On the ground, two ground control system (GCS) operate to command and acquire information. One is the data GCS that can show the state of the UAV and command the UAV to fly on waypoint or to achieve missions. The other is the vision GCS.

The vision GCS is built using LabVIEW. This provides the image from PNUAV, and has a command window where the user can input the angle of gimbal's pan and tilt or other control inputs. Command from the user input is transmitted to gimbal

controller through data GCS. Data GCS sends control inputs to move the vision system, as well as to maneuver the UAV.

3.1 Derivation of Arithmetic Range

In this paper, the dynamic relationship among the UAV, vision system, and GCS is derived. The 3-D position of the target is calculated from a monocular image as shown in Equations (1) and (2).

$$x = f\frac{X}{Z} \tag{1}$$

$$y = f\frac{Y}{Z} \tag{2}$$

Where, f is focal length, $(x, y)$is the position in the image, and $(X, Y, Z)$is the position in the real spacer. If the position along Z-axis is known, the 3-D information can be determined.

However, the 3-D information from Equations (1) and (2) is the result with respect to the camera frame. To transfer the coordinates from the camera to the reference frame, the attitudes of the camera and UAV are necessary. The attitude of a UAV with respect to a body fixed coordinate, is known as an onboard AHRS.

To get the position along Z, we assume the following ; 1) target is on a flat plane(zero altitude), and 2) the attitudes of UAV and camera can be measured. Using directional vector from the camera axes, the attitudes of the UAV and camera are obtained. A linear equation on 3-D is made from a unit directional vector from the camera to the target at zero altitude. The distance along Z-axis can be derived from a linear equation. Equations (3) and (4) represent the unit directional vector and the distance along Z, respectively [4].

$$\vec{l} = \left(T_V^i\right)^{-1}\left(T_C^V\right)^{-1}\begin{bmatrix}0 & 0 & 1\end{bmatrix}^T \tag{3}$$

$$\|\vec{m}\| = \sqrt{\left(\frac{l_1 z_v^i}{l_3}\right)^2 + \left(\frac{l_2 z_v^i}{l_3}\right)^2 + \left(z_v^i\right)^2} \tag{4}$$

Where, $\vec{l}$ is the unit directional vector between the camera and the target, T_V^i is the Euler transformation matrix of the UAV attitude with respect to the reference, and T_C^V is the Euler transformation matrix of the camera attitude with respect to the UAV. z_v^i is the altitude of the UAV with respect to the reference frame.

3.2 3-D Measurement using LPV

Method using arithmetic derivation can cause an error because of sensor noise. The result of this algorithm depends on the attitudes of the UAV and camera. This type of error is made at initial installation and by drift.

To reduce the error caused by the attitude sensor, the filter, which was studied, is applied on the 3-D measurement using monocular vision. The basic and popular filter is the Kalman filter for the estimation of the 3-D position. However, the dynamics derived to measure 3-D position of the target by monocular vision has nonlinearity. Although the Kalman filter can be applied on a nonlinear system, system error can be caused by the assumption of linearization. Therefore, the extended Kalman filter or a particle filter can be more effective. However, these filters are too complex to design and run. LPV, on the other hand, can be applied easily on the 3-D measurement of a target, and is effective on nonlinear systems.

General EOS uses a laser range finder or other measurement equipment to measure the distance between the camera and the target. Such additional equipment increases the weight and complexity of the system. Thus $\|\vec{m}\|$, which is derived for deciding the distance using the attitude and position of the camera and UAV, is used in this paper.

To improve the result in a asynchronous system, a modified nonlinear filter is derived. In [5], this filter is used on an out-of-sight problem. However, in this paper, filter is modified for an asynchronous system. Equation (5) represents the modified nonlinear filter.

$$\begin{cases} \dfrac{d}{dt}\left(\hat{P}_c\right) = -V + V_{tg} + sk_1 T_i^c H^{-1}(\hat{P}_c)(g_{\psi\theta}(\hat{P}_c) - y_m) \\ \dfrac{d}{dt}\left(\hat{V}_{tg}\right) = sk_2 T_i^c H^{-1}(\hat{P}_c)(g_{\psi\theta}(\hat{P}_c) - y_m) \\ \hat{P}_c = T_c^i(\hat{P}_c) \end{cases} \tag{5}$$

Where, V_{tg} and y_m are the position and measurement of the target, respectively. y_m is the same as $(g_{\varphi\theta}(P_c)$. $H(\hat{P}_c)$ is the Jacobian matrix of $g_{\varphi\theta}(\hat{P}_c)$. In Equation (5), data asynchronous means that non receiving UAV information and out-of-sight of target in image. Fig. 2 shows the results obtained with the modified nonlinear filter. Fig. 2, which are obtained from the modified filter system show good performance. The average distance error of the arithmetic method is about 94m, whereas the LPV is about 10m.

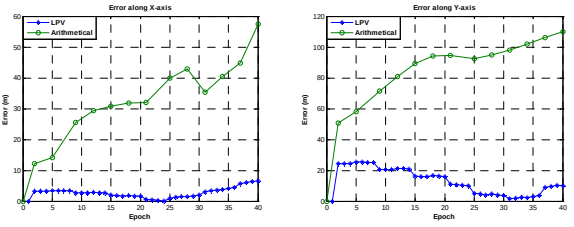


Fig. 2. Error along X-axis (left), Error along Y-axis (right).

4 Conclusions

This paper describes the development of a laboratory level vision system for small UAVs to measure the 3-D position of a target from a monocular image, acquired from the vision system on a UAV.

The Kalman filter is used to track the target in the image in real time. Through the algorithm for 3-D measurement, the 3-D position of the target can be estimated in the real space based on the target's position in the image, position and attitude of the UAV, and the attitude of the camera.

To make a simple vision system, the range between UAV and target is determined by arithmetic method. This range is used for a nonlinear filter system, derived from LPV and LMI. Since a nonlinear filter do not considered synchronism in receiving data, performance is similar with arithmetic method.

A modified nonlinear filter was suggested to prevent this problem. We can verify that the modified nonlinear filter is effective.

Acknowledgements

This research was financially supported by the Ministry of Education, Science Technology (MEST) and Korea Industrial Technology Foundation (KOTEF) through the Human Resource Training Project for Regional Innovation.

References

1. L. Merino, A. Ollero, J. Ferruz, J. R. Martinez-de-Dios and B. Arrue: Motion analysis and Geo-location for aerial monitoring in the COMETS multi-UAV system. in Proc. ICAR 2003 Coimbra (2003).
2. Klas Nordberg, Patrick Doherty, Gunnar Farneback, Per-Erik Forssen, Gosta Granlund, Anders Moe, Johan Wiklund: Vision for a UAV helicopter. Proceedings of IROS'02. workshop on aerial robotics (2002).
3. Milan Sonka, Vaclac Hlavac, Roger Boyle: Image Processing, Analysis, and Machine Vision. PWS Publishing. USA (1998).
4. Jong-hun Kim, Seon-yeong Jo, Jung-ho Kim, Dae-woo Lee, Kyeum-rae Cho, Ji-hwan Shin: Development of Laboratory Level Optical Module for Unmanned Aerial Vehicle using LabVIEW. 2008 Conference of Korea Institute of Military Science and Technology (2008).
5. Vladimir N. Dobrokhodiv, Isaac I. Kaminer, Kevin D. Jones, and Reza Ghabcheloo: Vision-Based Tracking and Motion Estimation for Moving targets using Small UAVs. Proceedings of the 2006 American Control Conference (2006).

Author Index

Proceedings of the 3rd International Workshop on
Intelligent Vehicle Controls & Intelligent
Transportation Systems - IVC & ITS 2009
ISBN: 978-989-674-003-0
http://www.icinco.org