The Analysis of Discrete-Event System in Autonomous Package Delivery using Legged Robot and Conveyor Belt

Garen Haddeler

***Abstract*— In this paper, the supervisory control of a Discrete Event System (DES) is analyzed to construct autonomous package delivery system. The delivery system includes legged robot in order to autonomously navigate uneven indoor terrain and a conveyor belt for transporting the package to the legged robot. The aim of the paper is using theory of supervisory control of DES to supervise and control machine's state and event and ensure robots autonomously collaborate. By applying the theory, we show collaboration of two individual robots to deliver goods in multi-floor environment The obtained results from the theory of supervisory control is implemented and verified in simulation environment.**

I. INTRODUCTION

Delivering package in indoor and uneven terrain can be challenging since today's robot cannot fully represent and navigate multi-storey terrain. Comparing with wheeled robots, legged robots can be used to navigate uneven terrains since legged robots can overcome larger obstacles than their body frame [1]. Inspired by the capability of such robots, we developed autonomous navigation framework for legged robot to fulfil desired behavior which for our case reaching goal, avoiding obstacles, climbing stairs and deliver goods. Moreover, conveyor belt is used to initially transfer goods to robot's body frame. Thus, robot and conveyor need to autonomously communicate each other and perform delivery. To do so, we developed supervisory control strategy to control individually machine's states and events and ensure robot and conveyor autonomously work together safely and efficient.

There are several autonomous robots project where used for package deliveries. Amazon's warehouses are actively using autonomous swarm robots which they delivery goods between one point to another using conveyor belts and wheeled robots [2]. However, main drawback of wheeled vehicles are goods can not deliver in multi storey facilities. To overcome this issue legged robot can be proposed instead of wheeled. A bipedal robot from Agility Robotics is used to deliver goods in multi-storey outdoor terrain [3]. Similarly, Boston Dynamic's wheeled-bipedal robot named Handle can perform delivery in indoor facilities [4]. However, their approach doesn't include collaboration with an conveyor belt and multiple robot aspect. By using Supervisory control theory for DES system, we propose autonomous collaboration between legged robot and conveyor belt.

The following parts of this paper are arranged as follows: Section II introduces the general concept of autonomous legged robot navigation and supervisory control theory. Thereafter, sub section III shows autonomous and system structure of a task based delivery scenario. Moreover, the proposed supervisory controlled structure is verified through simulation in Section IV-A. At the end, Section V concludes the work and gives an outlook on future research.

II. PRELIMINARIES

In this section, concept of legged robot and its autonomous navigation are presented. Moreover, some core methodologies of supervisory control and automata are introduced. This methodologies is used to construct our package delivery scenario.

A. Legged Robot

Autonomous navigation is needed in order to delivery goods from its starting position to desired goal position. Generally, in legged robots, self driving behavior can be carried out three sub-modules: robot localization, path planning and body controller.

1) Robot localization: Localization is one of the major step to perform autonomous navigation, path tracking, avoiding obstacles and mapping. Therefore, in this article, ready localization algorithm is used for navigation purposes. Since this isn't main focus of this work, we obtained robot's pose from simulation environment.

2) Path Planning: Path planning algorithm is used to generate path in between starting to goal positions and avoid static obstacles. We used 2D NavfnROS path planning algorithm to plan path at pre-defined route.

3) Body Controller: Comparing with wheeled robots, legged robot's body controller has higher complexity due to its dynamics and kinematics. One of the main reason is legged robots can be unstable while moving and needs to have robust controller in order to balance itself. We used MIT cheeta's body controller to balance its body frame, and perform foot placement from gait generator according to command reference (velocity). Note that the legged robot need to climb stairs therefore we use local awareness system to re-plan foot placements if location of planned foot is not traversable [1]. Additionally, local awareness' perception is obtained by traversability grid map where we evaluate terrain's slope and roughness information to detect step-able areas [5].

B. Concepts of automata and supervisory control

The supervisory control theory allows us to observe free events which are all possible combinations of events, and control controllable events so that agents fulfill given specifications [6].

Automata which is represented by Discrete event-states (DES) consists of a set of states and a set of events. An event causes a DES system to move from one state to another state and these events are assumed to occur instantaneously. Below formulation where states,Q, are represented by numbers and events $\sum$, are represented by symbols as following

$$Q = \{0 \quad 1 \quad 2\} \quad \sum = \{\alpha \quad \beta \quad \gamma\} \tag{1}$$

Automaton is constructed and represented as follow eq. 2 where Q is states, $\sum$ is events, δ transition function (which is relationship between current state, event and new state), q_0 indicates initial state and lastly, Q_m is marked states (which usually refers to system's equilibrium states).

$$G = \{Q \quad \sum \quad \delta \quad q_0 \quad Q_m\} \tag{2}$$

In order to present a sequence of the events, a string is defined as the sequence of symbols which may contain multiple symbols or one symbol. Language refers to the collection and combination of event and generated from automaton. An example of language and marked language can be shown as below eqs respectively 3, 4 . Note that marked language needs to satisfy $L_m(G) \subseteq L(G)$.

$$L(G) = \{\epsilon \quad \alpha\beta \quad \alpha\beta\gamma \quad \alpha\beta\alpha\gamma \quad ...\} \tag{3}$$

$$L_m(G) = \{\alpha\beta \quad \alpha\beta\gamma...\} \tag{4}$$

The supervisory control theory defines a supervisor function S which also called control policy so that it can control all controllable events of plant G where it follows given specification E. Controlled behavior represented as S/G and note that marked languages need to satisfy $L_m(S/G) \subseteq L_m(E)$ [6].

Briefly,by obtaining free behavior of plant G and defined specifications E, we can extract supervisor function S such that it can generate controlled DES and it's language $L_m(S/G)$. Further section III, we will show how we obtained our scenario's free behavior and it's specification.

III. Automation and System Structure

Our system has initially two real machines one robot and one conveyor. The delivery scenario starts with a legged robot spawning in first floor. Secondly, robot's first state is initialized by operator and it navigates through first goal which is in front of conveyor belt. Next, robot docks to belt and conveyor starts moving to transfer package to the robot's designated rectangular area. After conveyor belt finishes moving to place package, robot stands up. Then, by climbing stair, robot tries to reach second goal where is in one upper floor. Lastly, if robot reaches its second goal, robot change its state to success state and finishes its task until operator resend first goal.

To resolve task complexity in finite state machine wise, we divided our legged robot into two machine, therefore, our whole system consists of three sub-machines: First Task Robot, Conveyor Belt, Second Task Robot.

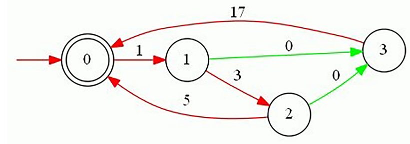

Fig. 1. The automaton model of Machine-1 First Task Robot: (Red) controllable, (Green) uncontrollable events

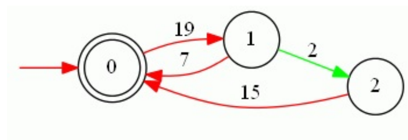

Fig. 2. The automaton model of Machine-2 First Task Robot: (Red) controllable, (Green) uncontrollable events

A. Machine-1: First Task Robot

Machine-1 is defined as reaching first goal using legged robot. Figure 1 shows states and events of the system using Automata. Table III-A defines states-events and their representations.

According to our delivery scenario, initially robot is in idle state which means robot is in stable-stopping position. An operator starts the delivery scenario by giving first goal to the robot. Thereafter walking state starts and legged robot uses proposed autonomous navigation framework to reach its first goal. After reaching its first goal, robot changes its state to docking and robot docks through conveyor belt. In this state, robot slowly approaches to belt and descends to collect package. After docking finished robot reaches idle state again and wait conveyor belt to transfer the package. As an uncontrollable event, if robot fails while walking or docking change its state to fail and by an error flag event robot's state becomes idle.

Table of State and Events: Machine-1 First Task Robot

States	Events
Robot Idle:0	First goal started:1
Walking:1	First goal reached:3, Robot failed:0
Docking:2	Docking finished:5, Robot failed:0
Fail:3	Error flag:17

B. Machine-2: Conveyor Belt

Machine 2 is defined as conveyor belt transferring goods to robot.Figure 2 shows states and events of the system using automata. Table III-B defines states-events and their representations.

According to our delivery scenario, conveyor belt is idle state until robot docking is finished. Thereafter, conveyor belt changes its state to working state and package is moved to robot's top. After package moving finished conveyor changes its state to idle again. As an uncontrollable event, if conveyor belt drops package to the ground in working state, it changes its state to fail state. Thereafter, box re-spawn on the conveyor belt, machine-2 change it's state to idle and whole process can be start again.

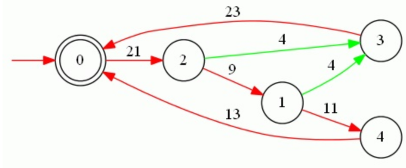

Fig. 3. The automaton model of Machine-3 First Task Robot: (Red) controllable, (Green) uncontrollable events

Table of State and Events: Machine-2 Conveyor Belt

States	Events
Conveyor Idle:0	Moving Box:19
Walking:1	Stopping box:7, Box dropped:2
Fail:2	Spawn box: 15

C. Machine-3: Second Task Robot

Machine-3 is defined as reaching second goal using legged robot. Comparing with machine-1, identical events and states represent different values. Figure 3 shows states and events of the system using automata. Table III-C defines states-events and their representations.

Similarly, with Machine-1, initially robot is in idle state which means robot is in stable-stopping position.Machine-3 start stand up state after package is on the robot's base. Thereafter walking state starts and legged robot uses proposed autonomous navigation framework to reach its second goal where is one level above from pick-up position.After reaching its second goal, robot changes its state to success and later, by a success flag, it changes to idle state in order to finish it's task and stop until operator gives first goal again. As an uncontrollable event, similarly with Machine-1, if robot fails while walking or standing up change its state to fail and by an error flag event, robot's state becomes idle

Table of State and Events: Machine-3 Second Task Robot

States	Events
Robot Idle:0	Box is on the robot:21
Walking:1	Second goal reached:11, Robot failed:3
Stand up:2	Second goal started:9, Robot failed:3
Fail:3	Error flag:23
Success:4	Success flag:13

D. Specifications

According to Supervisory theory, specifications are needed to control free behavior of the system. To do so, we have defined 8 specification as follows:

1) After docking finished (5), moving box (19) event must start.
2) After moving box (19), stopping box (7) event must start or as an uncontrollable event box can be dropped (2)
3) After stopping box (7), box is on the robot (21) must start.
4) After box is on the robot (21), second goal started (9) event must start or as an uncontrollable event robot can be failed (4).

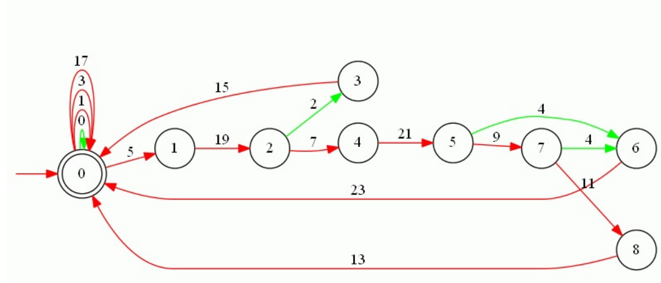

Fig. 4. The automaton model of Specifications: (Red) controllable, (Green) uncontrollable events

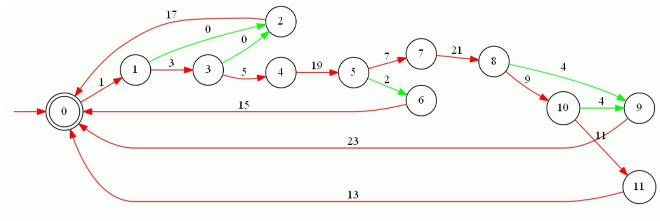

Fig. 5. The automaton model of Controlled behavior: (Red) controllable, (Green) uncontrollable events

5) After second goal started (9), second goal reached (11) event must start or as an uncontrollable event robot can be failed (4).
6) After second goal reached (11), success flag (13) event must start
7) After uncontrollable event robot fails (0 and 4), error flag (23) event must start.
8) After uncontrollable event box dropped (2), spawn box (15) event must start.

IV. Experimental

We obtained our DES Supervisor using TCT software and execute autonomous delivery scenario in physical simulator.

A. TCT Software

This program allows us to synthesis of supervisory controls for discrete-event systems. In this section, firstly we combined eight specifications into one specification represented as E, which is done by using TCT Software's "MEET" function. Following Figure 4 can be obtained as representation of finite state and events system. Secondly, the synchronized combination of Machine-1,Machine-2 and Machine-3 are obtained from using TCT software's "SYNC" function. The results of sequences, which is free behavior of delivery package scenario, shown in Figure- 8. As it can be observed that, states and events are massive which includes 60 states and 254 transitions. These states and events contain all the possibility of control sequences which we call free behavior or plant G.

Lastly, to obtain controlled behavior, supervisor function (S) can be calculated out by using TCT software's "SUPCON" function which uses given specifications (E) to control plant (G). Thereafter it returns a non-blocking, minimally restrictive supervisor S/G. Controlled behavior of the

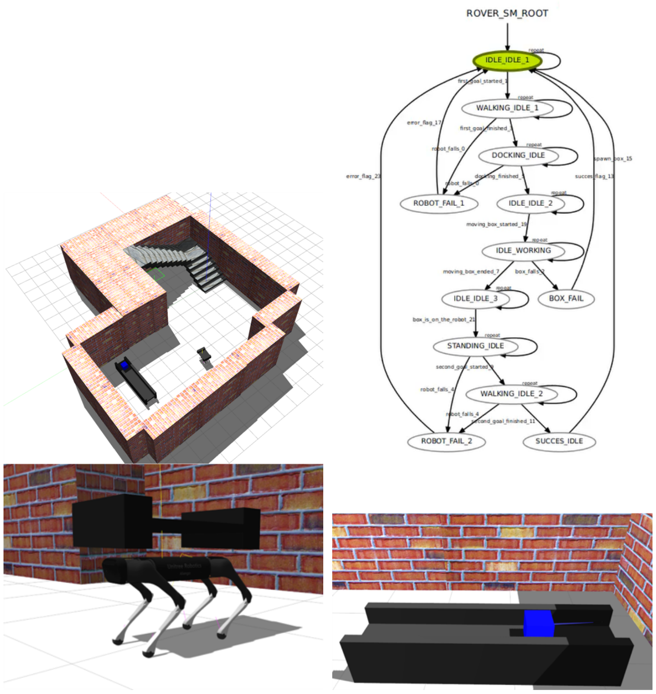

Fig. 6. Simulation Environment in Gazebo: (Up-Left) indoor-uneven environment which includes stair, (Up-Right) states and events in ROS-Smach implementation (Bottom Left) Quadrupedal robot which package carrier is mounted it's back, (Bottom Right) Conveyor belt which moves box to robot's back

autonomous delivery package scenario can be shown Figure 5. As one of the controlled behavior's states sequence (0-1-3-4-5-7-8-10-11) can confirm that, our controlled behavior meets with specifications and perform its task accordingly.

B. ROS-Gazebo Simulator

We have modelled our autonomous package delivery scenario in ROS-Gazebo environment. Our simulation environment similarly includes high walls and stairs which is shown Figure 7-up-left. The controlled behavior's DES implemented to ROS-Smach and visualized as Figure 7-up-right. Quadrupedal robot is represented as Figure 7-bottom-left and conveyor belt is shown as Figure 7-bottom-right.

Furthermore, we used ROS-Smach library to represent controlled behavior in aspect of finite state and event system. Leg Locomotion, path planning and smach algorithms are worked separately in one ROS environment. All events are subscribed from ROS environment to perform action and all events published to ROS environment in order to change its current state.

C. Result

In result section, we obtained controlled DES behavior of autonomous package delivery scenario and fully functional simulation environment where conveyor belt moves box and legged robot navigate through desired goal positions. It was observed that by implementing controlled behavior of finite state and events, robot is able to take package from conveyor and deliver-climb to specific location where is one floor above.

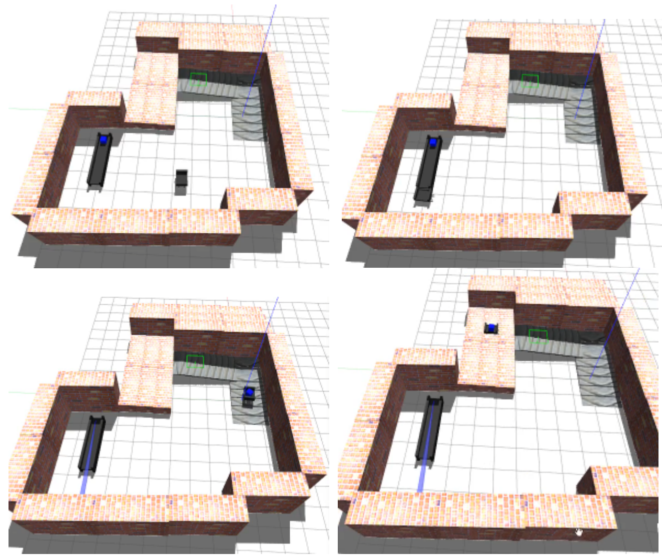

Fig. 7. Snapshot of Autonomous Package Delivery Scenario:(Up-Right) initial Position , (Up-Left) first goal reached, (Bottom-Left) climbing to the stairs, (Bottom-Right) second goal reached

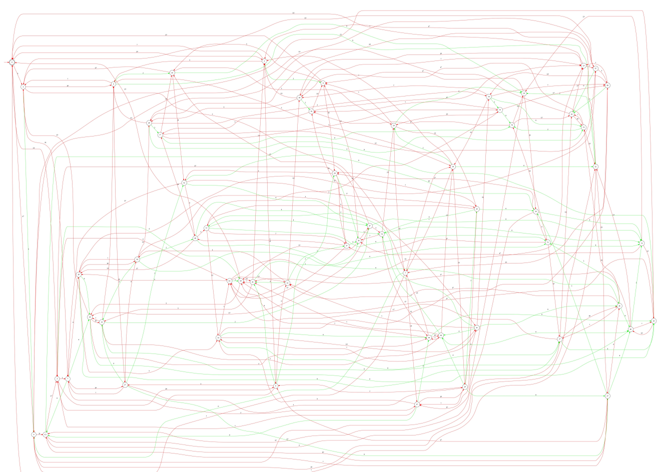

Fig. 8. The automaton model of Free behavior

V. Conclusion

In this paper, supervisory control of DES is implemented to analyze the acceptable control sequence among free sequence for autonomous package delivery scenario using legged robot and a conveyor belt. After the defining DES for each machine and their specification, automated model of controlled behavior is obtained. Thereafter, to show effectiveness of supervisory control theory, DES of controlled behavior is implemented on simulation environment and it successfully perform autonomous package delivery in multi-storey terrain. For future work, we can obtain controlled DES model of multiple legged robots and conveyors and test in simulation and real world.

References

[1] W. Bosworth, S. Kim, and N. Hogan, "The mit super mini cheetah: A small, low-cost quadrupedal robot for dynamic locomotion," *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2015.

[2] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, "Swarm robotic behaviors and current applications," *Frontiers in Robotics and AI*, vol. 7, 04 2020.

[3] Y. Gong, R. Hartley, X. Da, A. Hereid, O. Harib, J. Huang, and J. W. Grizzle, "Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway," *CoRR*, vol. abs/1809.07279, 2018. [Online]. Available: http://arxiv.org/abs/1809.07279

[4] M. Bjelonic, C. D. Bellicoso, Y. D. Viragh, D. Sako, F. D. Tresoldi, F. Jenelten, and M. Hutter, "Keep rollin'—whole-body motion control and planning for wheeled quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, p. 2116–2123, 2019.

[5] M. Wermelinger, P. Fankhauser, R. Diethelm, P. Krusi, R. Siegwart, and M. Hutter, "Navigation planning for legged robots in challenging terrain," *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[6] P. J. G. Ramadge and W. M. Wonham, "The control of discrete event systems," *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–98, 1989.