The Global Dispatcher

Richard Peters

Peters Research Ltd., Bridge House, Station Approach, Great Missenden, Bucks, HP16 9AZ, UK

Keywords: traffic control, dispatching, global, universal, conventional control, destination control, single deck, double deck.

Abstract. A modern lift traffic control system, often known as a dispatcher, can collect passenger calls in several ways. Conventional dispatching uses up-and-down buttons on the landing with additional buttons for each floor in the car. Destination control dispatching uses destination input devices on the landings so that passengers can select their required floor when the lift is first called. Hybrid dispatching systems use a combination of landing call buttons, car call buttons and destination input devices. Aside from a range of input devices, advanced dispatchers may manage single and double-deck lifts, multiple lifts in the same shaft, and a combination of these lift types within a lift group. This paper describes how the same dispatching software can manage all input devices and lift types by applying a unified approach to lift dispatching. The core software is built on a lift controller software model, which can manage all lift and call types. Unknown information, for example, future car calls once a landing call is answered, is predicted. The choice of which lift serves which call is made by applying a simulation model, which assesses the outcome of alternative allocations the dispatcher could make. The Global Dispatcher applies the Global Dispatcher Interface.

1 INTRODUCTION

Lift dispatching describes managing and coordinating the movement of lifts within a vertical transportation system to transport passengers and goods efficiently between different floors. Lift dispatching plays a pivotal role in managing the efficient movement of passengers within multi-story buildings.

Conventional dispatching uses up-and-down buttons on the landings with additional buttons for each floor in the car. Destination control dispatching uses destination input devices on the landings so that passengers can select their required floor when the lift is first called. Hybrid dispatching systems use a combination of landing call buttons, car call buttons and destination input devices. Aside from a range of input devices, advanced dispatchers may manage single and double-deck lifts, multiple lifts in the same shaft, and realistic combinations of these lift types within a lift group.

The objective of the Global Dispatcher is to provide a unified framework so that all realistic options can be addressed within a single set of dispatcher software. This paper sets out an approach to solving this problem.

2 THE CONTROLLER

2.1 About the lift controller

Lift controllers are responsible for managing the movement and behaviour of lifts in a building. There is one lift controller per lift. The primary purpose of a lift controller is to ensure safe, efficient, and smooth transportation of passengers or goods between different floors.

The lift controller monitors various inputs, such as button presses inside the elevator car and on the floors, door status, and car position. Based on these inputs and the pre-programmed logic, the controller determines the appropriate actions, such as opening or closing the doors, stopping at a floor, and travelling to the desired destination.

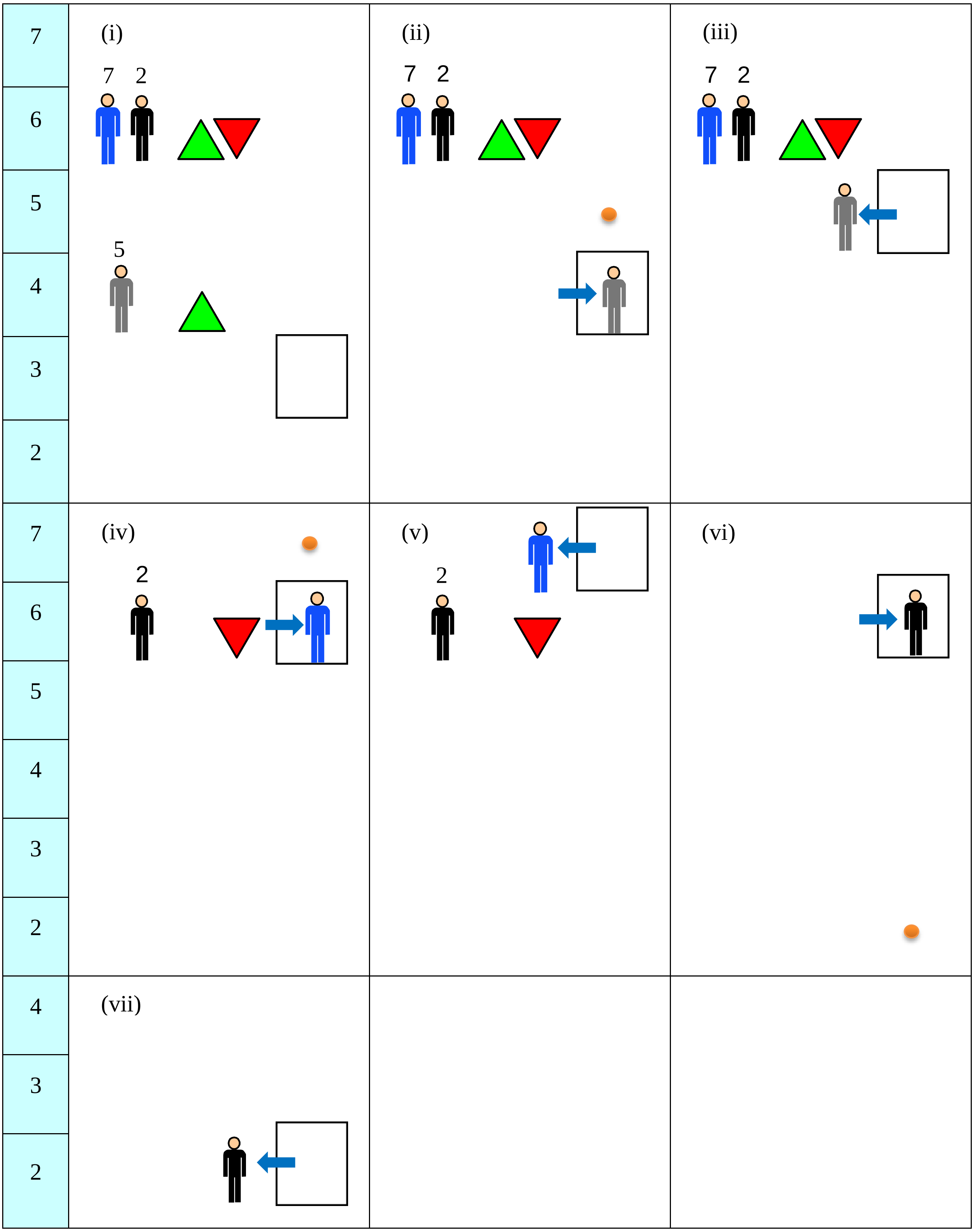

Figure 1 Collective operation (passenger destinations shown above heads)

2.2 Collective control

A single (simplex) lift does not need a dispatcher; its controller includes software enabling it to serve its calls logically. Most modern lifts answer calls collectively, as illustrated in Figure 1. All landing calls and the resulting car calls in one direction are served; then, the car reverses and serves calls in the opposite direction.

For lift groups, a simple approach to dispatching is to have separate dispatching software to decide which lift will serve which call. The inputs are the status of the lifts and the calls registered. The outputs are which lifts should serve which landing and/or destination call. The dispatcher software must also understand collective control to assess how long each lift would take to answer a call. Once the allocation is made, the call can be passed from the dispatcher to the selected lift, adding it to its schedule according to collective control rules.

The problem with this approach is that all lift controllers implement collective control slightly differently, and the assumptions of the dispatcher and controller can conflict. For example, consider the scenario illustrated in Figure 2 (i) where a lift is travelling to serve a down call. While the lift is travelling to the down call, a new call is registered travelling up from the same landing; see Figure 2 (ii). The dispatcher cannot, with confidence, predict whether the controller will serve the up or down call first.

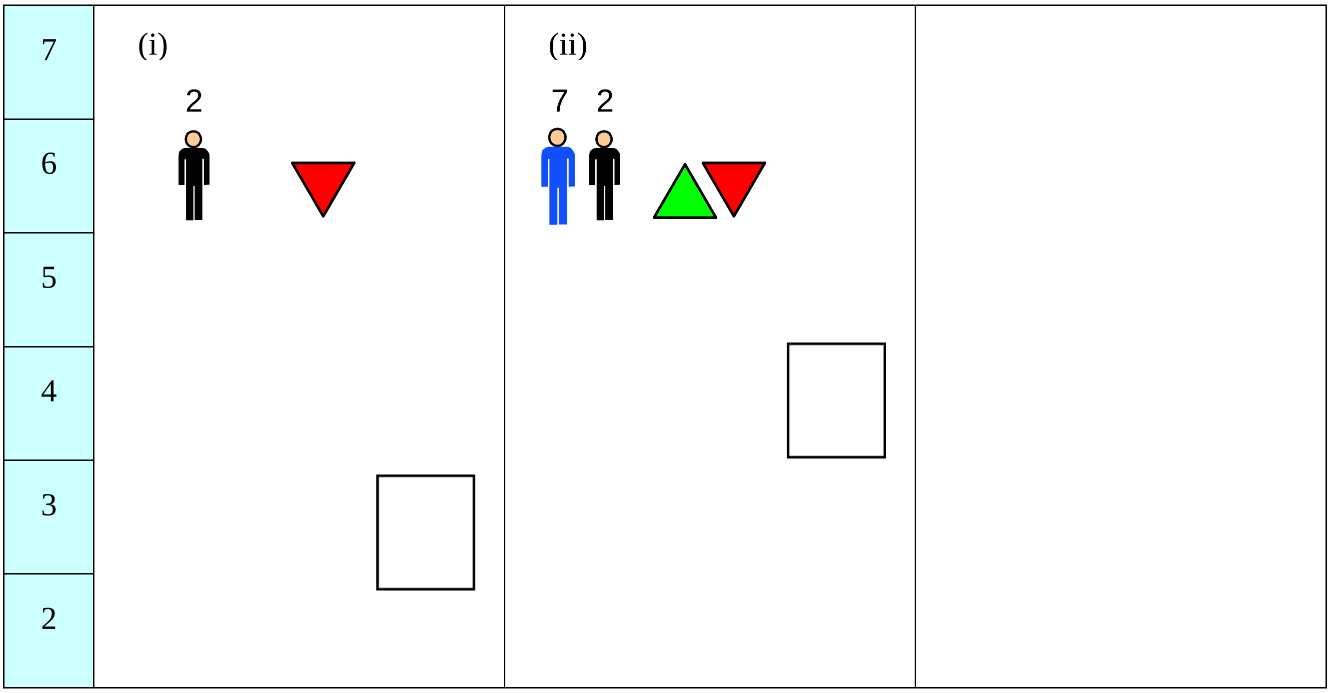

Figure 2 Collective control operational inconsistencies – which call will be served first?

There are also scenarios where the dispatcher might not want to be restricted by the controller's logic. For example, if both passengers had loaded in Figure 1 step (iv), the stop at step (vi) could have been avoided, saving time. A passenger would be taken in the "wrong direction" at first, which is considered unacceptable [1]. Nevertheless, with destination control, these so-called "reverse journeys" can be advantageous [2] and may be a dispatcher option.

Thus, for a Global Dispatcher capable of working with a range of lift controllers, the collective control logic is best removed from the lift controller and implemented solely as part of the dispatcher function; the Global Dispatcher is then responsible for instructing the lift on its destination floors, one at a time, rather than allocating calls to the lift controller implementing its own version of collective control.

2.3 Door operation

As for the collective control logic, the lift controller has historically managed door operation. This limits the system's overall efficiency as the dispatcher has additional information, which can save time. For example, in some instances, the dispatcher knows that just one person is loading or unloading the car. Once the door beams have been broken and reestablished, an intelligent dispatcher will know the doors can be closed immediately rather than waiting for a lift controller's dwell time to expire. For safety reasons, the lift controller must remain in charge of opening and closing the doors, but the Global Dispatcher can provide the logic to send door open and close requests. To do this, the dispatcher must be provided (by the controller) with door beam status so it knows when passenger transfer begins and ends.

3 COLLECTIVE CONTROL WITHIN THE GLOBAL DISPATCHER

3.1 Managing call types

The collective control function within the Global Dispatcher needs to manage collective control for both destination calls and conventional calls generated by passengers, see section 7. The mixture of call types is possible as a destination call is equivalent to a landing call with a car call [3]. When a landing call is registered, it can be treated as a destination call with an inferred car call. Once a landing call is answered and any new car calls have been registered, the destination call can be updated. For example, a down call from level 7 may initially be treated as a destination call from level 7 → ground floor. When the down call is answered, and a car call to level 1 is registered, that destination call may be updated from level 7 → ground to level 7 → level 1. In some instances, for example, using load weighing and when multiple car calls are registered, one landing call may be assumed to spawn multiple destination calls; for example, a down landing call from level 7 may initially be treated as a single destination call from level 7 → ground floor. If the load weighing demonstrates two extra passengers load at level 7, and then car calls to levels 3 and 4 are registered, the single destination call level 7 → ground may be replaced by two destination calls, level 7 → level 4 and level 7 → level 3. Aside from enabling call types to be mixed, this approach brings passenger-centric dispatching to conventional control, see section 4.1.

3.2 Lift types

The collective control function is designed to manage double-deck lifts; a single-deck lift is treated as a double-deck lift but with a block on any allocations to the (non-existent) upper deck. Two lifts per shaft are implemented based on single-deck lifts with an additional rule set, see section 6.

3.3 Application of the collective control function

The collective control function is applied within the Global Dispatcher in two different contexts:

1. To model the movements of the lift, determining the probable outcome if a call is allocated to a lift. This helps the dispatcher to choose the "best" lift to allocate a call to.
2. To determine where to send the lift to and door operation once calls have been allocated to lifts.

In the second context, controlling the actual lift, the real system provides the door and door beam status, which changes due to real passengers loading and unloading a lift. In the first context, an internal simulation, the simulation code provides the door and door beam status, mimicking the effect of passengers based on its knowledge of the registered calls.

4 OPTIMISATION GOALS AND ALLOCATING LIFTS

4.1 System versus passenger-centric objectives

There are many approaches to allocating calls to lifts. A common strategy for conventional control is to assess the Estimated Time of Arrival (ETA), i.e., how long would it take lift A, lift B, and lift C to answer a new landing call. The lift with the lowest ETA is allocated the call. There may be other considerations; for example, a “co-incident call bonus” may be applied to reduce the ETA and make it more likely that lifts already stopping at the landing call floor receive the allocation [4].

Many dispatchers use these and other system-based measures to choose the “best lift”. A more sophisticated approach is to make the assessment passenger-centric. This is the basis of most destination control algorithms [1] [3], which consider every passenger's waiting and transit time. Passenger-centric optimisation goals can also be applied to conventional and hybrid control if conventional calls are translated to destination calls (see section 3.1) before the allocation process. The internal simulation of the Global Dispatcher only needs to recognise destination calls.

By making the objectives passenger-centric, any practical combination of destination and conventional landing calls can be allocated for any lift type (single-deck, double-deck, two cars per shaft).

4.2 Optimisation algorithms

The objectives are implemented in an optimisation algorithm which translates each “passenger experience”, as determined by the internal simulation, into a cost. The increase in total cost arising from an allocation of a call to a lift can be determined, and the lift with the lowest cost is allocated.

An optimisation algorithm based on minimising the total time to destination is widely applied. Still, research into the psychology of waiting suggests that different parts of the journey may be more frustrating than others [5] [6].

To address this, the optimisation algorithm can account for a range of human factors. Indeed, it is possible to optimise on anything that can be modelled, including energy consumption [7].

Specific to double-deck lifts and systems with two lifts per shaft, a passenger’s journey may be delayed by unseen activity in another lift or cabin delaying a passenger’s journey. This can be addressed by breaking up the lift journey into more phases, including departure delay and blind departure delay [8]. If the relative frustration of each phase of the lift journey can be qualified, this can also be part of the optimisation algorithm.

The disproportionate “pain” of long waits can be accounted for if each extra second is weighted more than the previous [7].

The application of Artificial Intelligence is beyond the scope of this paper, but for an introduction, refer to [4].

5 DOUBLE-DECK LIFTS

Double-deck lifts have two cabs in one unit, serving adjacent floors together. Escalators connect the lower and upper ground floors. The technology is available from most major suppliers.

Double-deck lifts are most efficient if, during peaks, lower cabs serve odd floors and upper cabs serve even floors. With destination control, this restriction is easy to avoid in the software, although it can significantly impact handling capacity and the overall quality of service.

The Global Dispatcher implementation is based on applying the collective control function. There are restrictions aside from the lower cab not being able to serve the top floor and vice-versa; some allocations must be banned.

5.1 Banned allocations due to passenger loading the wrong car

In Figure 3, a passenger is loaded into the upper cab travelling to level 6. There is a passenger at level 6 wanting to travel to level 8. The dispatcher must consider both lower and upper cab allocation possibilities. If the lower cab is allocated, the upper cab will stop at level 6 first and the lower cab second. The person will get into the upper cab instead of the lower cab by mistake. So, the allocation of the passenger to the lower cab must be banned.

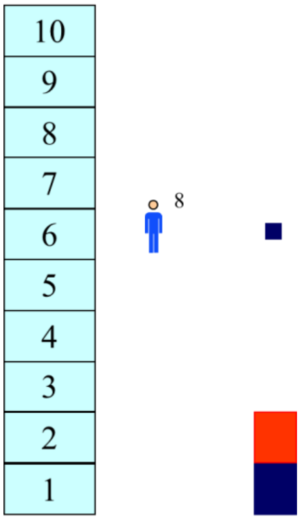

Figure 3 Scenario where person loads the wrong car

6 TWO CARS PER SHAFT

6.1 General

Installations with two cars per shaft are currently only available from one supplier [9]. Their handling capacity limit is comparable to double-deck lifts, which is apparent if you consider them as a double-deck lift with an additional degree of freedom, i.e., the two cabins are not attached to each other [10].

Managing two lifts in one shaft introduces unique dispatching challenges as follows. The core Global Dispatcher group collective algorithm must be supplemented by additional logic to avoid collision, impasse scenarios, and passengers getting into the wrong car.

6.2 Collision avoidance

The first step required is to have a collision avoidance strategy; for example, see Figure 4 from [7]. This involves holding back one car where the next destination cannot be completed without collision and inserting additional calls to move an obstructing car out of the way.

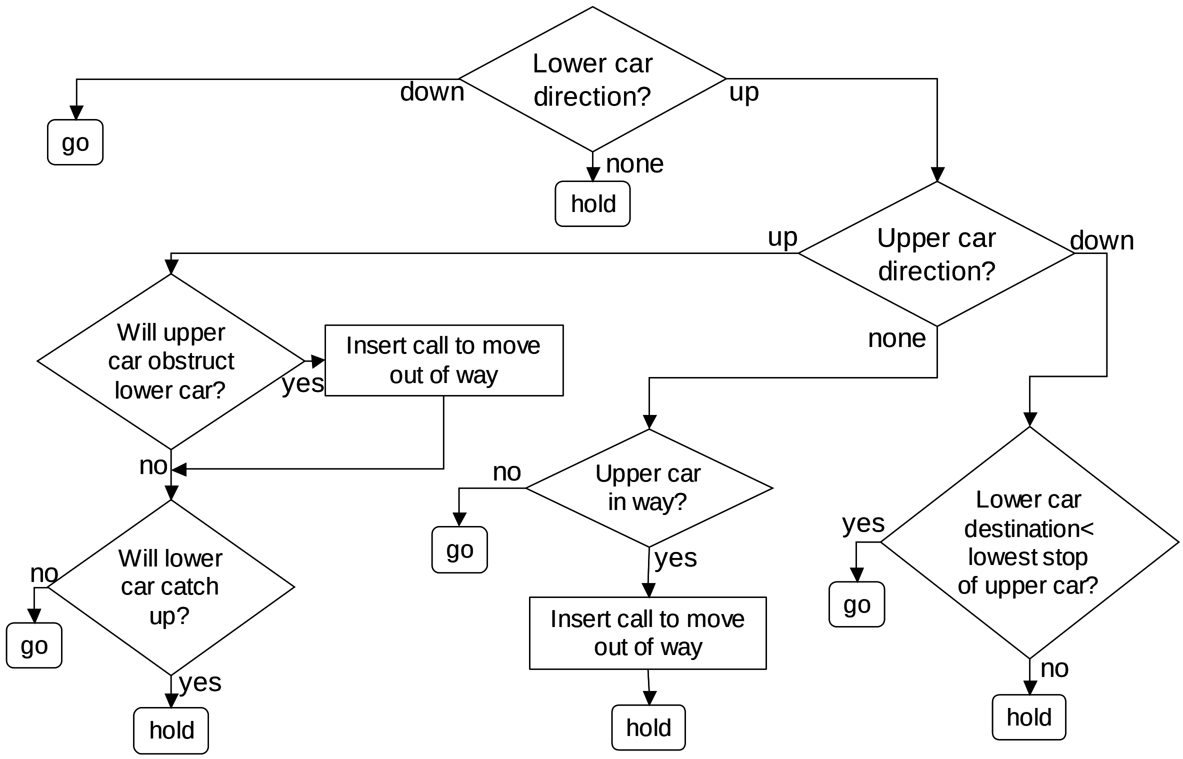

Figure 4 Collision avoidance strategy

6.3 Banned allocations due to impasse scenarios

Another issue is that some combinations of calls result in an impasse; this occurs when to complete the allocated calls, we must reverse a car which already has a passenger in it, see Figure 5.

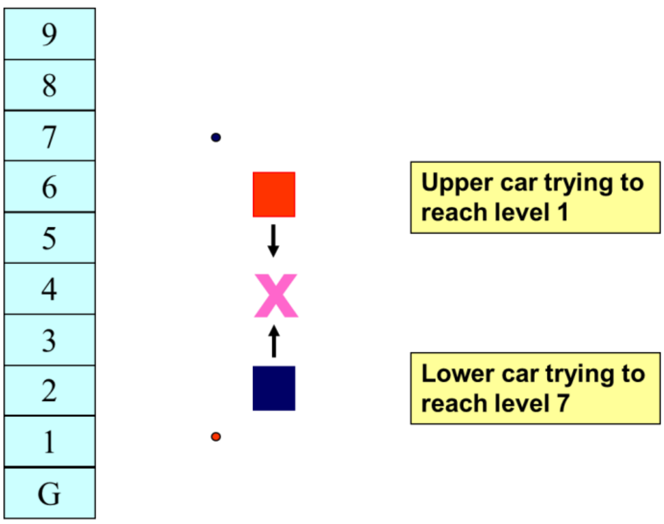

Figure 5 Impasse scenario

6.4 Banned allocations due to passenger loading the wrong car

Like double-deck lifts, there is a risk that a passenger will get into the wrong car. In the example represented in Figure 6, the dispatcher wants to consider allocating a call from level 7 to the lower car. However, the upper car is already scheduled to stop at level 7. As the two cars are both loaded from the same landing doors, the passenger that the dispatcher wants to load the lower car is likely to load the upper car by mistake.

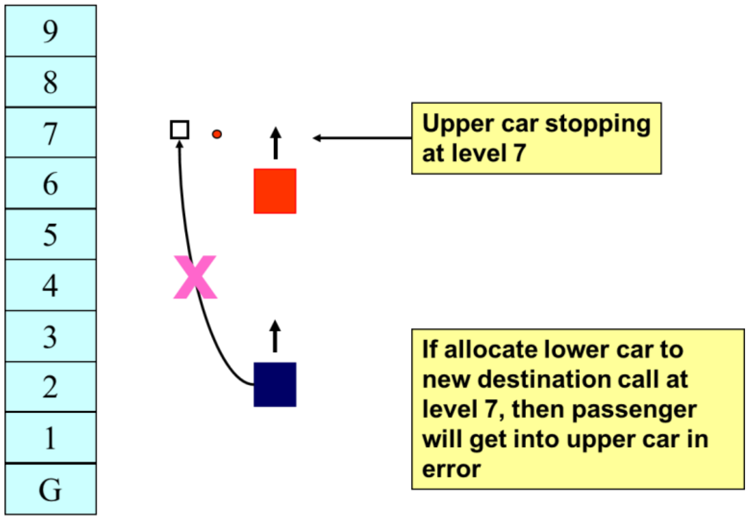

Figure 6 Scenario where person loads the wrong car

7 IMPLEMENTATION

The Global Dispatcher applies the Global Dispatcher Interface [11]. Messages are communicated over TCP/IP applying Protocol buffers, which is a language-neutral, platform-neutral, extensible mechanism for serializing structured data.

All the dispatching calculations are implemented in a single software module (a self-contained and reusable unit of code). In simulation [12] this software module runs in parallel to the simulation software on the user's computer. The same software module is run in buildings on an Industrial IoT Edge Gateway device [13] to direct lift controllers [14].

The Global Dispatcher applies central (rather than distributed) control as part of the group controller, see Figure 7. All the dispatching calculations take place in the group controller, which is required to achieve the benefit of a modular, controller-independent dispatcher. In the alternative, a distributed system, each lift controller is responsible for calculating a "bid" for the new call and any one of the lift controllers can act as a master to review the bids and make the allocation. This distributed approach is inherently robust, something central control relies on a backup device to achieve.

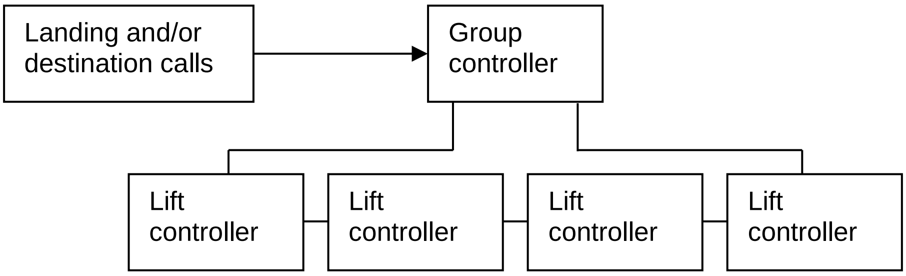

Figure 7 Centralised control (from Figure 9.3 CIBSE Guide D: 2020 [15])

Destination input devices are off-the-shelf Android kiosk tablets with Power-over-Ethernet [16], running a non-proprietary kiosk browser [17]. Conventional landing and car calls require additional hardware to convert button presses to messages on the same network. All software updates and configuration options can be applied remotely from off-site.

8 CONCLUSIONS

The conception of the Global Dispatcher and its evolution emerged from a culmination of thirty-six years engaged in the design and subsequent deployment of dispatcher algorithms.

In that time, the complexity of dispatching has grown with the introduction of diverse input devices and lift configurations. Within our proprietary simulation software [12], we have developed and continue to manage fourteen such dispatchers. While retaining the need to employ legacy dispatchers for assessing modernisation projects, there also exists a requirement for a more optimised resolution to this challenge.

Another challenge encountered has been the necessity of formulating distinct dispatchers for varying controllers. By transferring certain decisions typically made by the controller to the dispatcher, a singular dispatcher software module can be universally applied across all lift controllers.

Furthermore, by applying passenger-centric optimisation objectives and treating conventional landing calls as destination calls with inferred destinations, the Global Dispatcher can evaluate all feasible allocations across different lift configurations under a uniform criterion. This capability empowers the dispatcher to impartially juxtapose calls from diverse input devices against potential assignments to various lift types. While the simultaneous deployment of single-deck, double-deck, and two-cars-per-shaft solutions within the same group is improbable, this capability facilitates evaluating and deploying all conceivable combinations of such.

The Global Dispatcher remains an ongoing undertaking. While not all functions have been fully realised and deployed, the foundational framework and concepts have been substantiated. Preliminary iterations are operational both in simulation and real-world building scenarios.

REFERENCES

[1] G. Barney and L. Al-Sharif, Elevator Traffic Handbook, London, 2015.

[2] S. Gerstenmeyer and R. Peters, "Reverse Journeys and Destination Control," in *Proceedings of the 4th Symposium on Lift and Escalator Technologies*, Northampton, 2014.

[3] R. Smith and R. Peters, "ETD Algorithm with Destination Control and Booster Options," in *Elevator Technology 12*, The International Association of Elevator Engineers, 2002.

[4] R. Peters, "Elevator Dispatching," in *Proceedings of ELEVCON 2014*, Paris, 2014.

[5] C. Bird, R. Peters, E. Evans and S. Gerstenmeyer, "Your Lift Journey – How Long Will You Wait?," in *Proceedings of the 6th Symposium on Lift and Escalator Technologies*, Northampton, 2016.

[6] R. Smith and S. Gerstenmeyer, "A review of Waiting Time, Journey Time and Quality of Service," in *Proceedings of the 3rd Symposium on Lift and Escalator Technologies*, Northampton, 2013.

[7] R. Smith and R. Peters, "Enhancements to the ETD Dispatcher Algorithm," in *Elevator Technology 14*, The International Association of Elevator Engineers, 2004.

[8] S. Gerstenmeyer, R. Peters and R. Smith, "Departure Delays in Lift Systems," in *7th Symposium on Lift and Escalator Technologies*, Northampton, 2017.

[9] TK Elevator, "TWIN - elevator by TKE," TK Elevator, [Online]. Available: https://www.tkelevator.com/uk-en/products/elevators/twin/. [Accessed 10 08 2023].

[10] R. Peters, "Lift Planning for High-Rise Buildings," in *8th Symposium on Lift & Escalator Technology* , Northampton , 2018.

[11] R. Peters, "Global Dispatcher Interface," in *Proceedings of the 6th Symposium on Lift and Escalator Technologies*, Northampton, 2016.

[12] Peters Research Ltd, "Elevate - Peters Research," Peters Research , [Online]. Available: https://peters-research.com/index.php/elevate/. [Accessed 10 08 2023].

[13] Embedded Micro Technology, "Industrial IoT Edge Gateway Raspberry Pi Compute Module," Embedded Micro Technology, [Online]. Available: https://www.embeddedpi.com/industrial-iot-edge-gateway. [Accessed 10 08 2023].

[14] Lester Controls Limited, "Lift Controllers - Controllers Lester Control Systems," Lester Controls Limited, [Online]. Available: https://www.lestercontrols.co.uk/en/controllers/lift-controllers.html. [Accessed 10 08 2023].

[15] R. Peters, "Lift traffic control," in *CIBSE Guide D:2020 Transportation systems in Buildings*, London, The Chartered Institution of Building Services Engineers, 2020, pp. 9-1 to 9-28.

[16] ProDVX Europe Office, "APPC-10XPL | ProDVX," ProDVX Europe Office, [Online]. Available: https://www.prodvx.com/products/appc-10xpl. [Accessed 10 08 2023].

[17] Fully Factory GmbH, "Fully Kiosk Browser Lockdown | Android Kiosk Mode App," Fully Factory GmbH, [Online]. Available: https://www.fully-kiosk.com/. [Accessed 10 08 2023].

ACKNOWLEDGEMENTS

The author would like to acknowledge Dr Mike Pentney of Peters Research Ltd and Mr Brendan Johnson of Lester Controls Ltd without whom the Global Dispatcher and the Global Dispatcher Interface would have remained a theoretical concept.

BIOGRAPHICAL DETAILS

Richard Peters has a degree in Electrical Engineering and a Doctorate for research in Vertical Transportation. He is a director of Peters Research Ltd and a Visiting Professor at the University of Northampton. He has been awarded Fellowship of the Institution of Engineering and Technology, and of the Chartered Institution of Building Services Engineers. Dr Peters is the principal author of Elevate, elevator traffic analysis and simulation software.