CrossMark

Drone delivery systems: job assignment and dimensioning

Pasquale Grippa[1] · Doris A. Behrens[2,3] · Friederike Wall[4] · Christian Bettstetter[1,5]

Abstract

This article studies how to dimension and control at the system level a fleet of autonomous aerial vehicles delivering goods from depots to customers. Customer requests (jobs) arrive according to a space-time stochastic process. We compute a lower bound for the infrastructure expenditure required to achieve a certain expected delivery time. It is shown that job assignment policies can exhibit a tipping point behavior: One vehicle makes the difference between almost optimal delivery time and instability. This phenomenon calls for careful dimensioning of the system. We thus demonstrate the trade-off between financial costs and service quality. We propose a policy that assigns each incoming job to the vehicle that will do the job faster than other ones, seeking to minimize the overall workload in the system in the long term. This policy is scalable with the number of depots and vehicles, performs optimal in low load, and works well up to high loads. Simulations suggest that it stabilizes the system for any load if the number of vehicles per depot is sufficient.

Keywords Unmanned aerial vehicles (UAVs) · Queuing theory · Task allocation · Decision making · System dimensioning · Level of service · Investment

This is one of several papers published in *Autonomous Robots* comprising the "Special Issue on Robotics Science and Systems".

✉ Pasquale Grippa
pasquale.grippa@aau.at

Doris A. Behrens
BehrensD1@cardiff.ac.uk

Friederike Wall
friederike.wall@aau.at

Christian Bettstetter
christian.bettstetter@aau.at

1 Institute of Networked and Embedded Systems, Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria

2 School of Mathematics, Cardiff University, 23 Senghennydd Road, Cardiff CF24 4AG, UK

3 Aneurin Bevan University Health Board, Lodge Road, Caerleon NP18 3XQ, UK

4 Institute for Business Management, Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria

5 Lakeside Labs GmbH, Klagenfurt, Austria

1 Introduction

Small drones, also called unmanned aerial vehicles (UAVs), have successfully made their way to civil applications. They fly routes in an autonomous manner, carry cameras for aerial photography, and transport goods from one place to another. The range of applications is broad, including aerial monitoring of plants and agricultural fields as well as support for first time responders in case of disasters (Andre et al. 2014; Kovacina et al. 2002; Lima et al. 2014; Quaritsch et al. 2008). Delivering goods with a fleet of drones (Grippa 2016; Grippa et al. 2017) becomes an option if classical means of transportation—such as trucks, trains, and planes—are inappropriate. This comes about if roads, railway tracks, or landing facilities do not exist, if weather conditions make it impossible to use them, or if their use is too dangerous or time consuming. A compelling service in this context is the delivery of medicine, vaccinations, or laboratory samples for patients in remote areas and crisis regions. Such a service is also worthwhile in densely populated metropolitan areas when congestion makes roads nearly impassable.

This article investigates drone-based delivery at a system level. The entities of the system are goods, customers, vehicles, and depots. Customers request goods that are stored in depots and delivered by vehicles. Service requests, also denoted as jobs or customer requests, are not known in

advance and arrive over time at certain locations according to a space-time stochastic process. We introduce an approach to dimension the system (How many vehicles and depots are needed for a certain area?) and propose and analyze policies for job assignment (How to assign customer requests to vehicles to minimize the expected delivery time?).

If the vehicles are capable of consecutively serving several customers before they return to a depot, e.g., if goods are lightweight and total distances are small, the problem of selecting customer requests to be served falls into the domain of dynamic vehicle routing with stochastic demands, dating back to Bertsimas and Ryzin (1991, 1993a, b). In contrast to this, vehicles in our system serve no more than one customer per trip for capacity reasons, and we use the term *job assignment* to emphasize the difference to routing. Focusing on depot-to-customer delivery, we complement research on routing policies for wide-area surveillance (Bullo et al. 2011; Enright et al. 2009; Frazzoli and Bullo 2004; Pavone et al. 2011; Savla et al. 2008).

We model the system as an $M/G/K$ queue with $K > 1$. Our performance measure is delivery time, i.e., the time it takes for a customer from requesting to obtaining a good. The stability of this service is linked to the queuing of jobs, i.e., customers may have to wait until other customers have been served. The system becomes unstable if the average number of waiting customers persistently increases over time.

Our contributions are as follows: First, we give a lower bound for the expenditure needed to set up a stable system as a function of the targeted average delivery time. Second, we analyze two simple job assignment policies: *Nearest job first to random vehicle* (NJR) and *first job first to nearest vehicle* (FJN). These policies are suboptimal but their performance yields insight on the system dynamics which helps us to design advanced policies. Third, we analyze a policy known from queuing theory, named *first job to vehicle with the smallest workload* (FJWπ) in this paper, and modify it according to the lessons learned from NJR and FJN. We propose the *first job first to vehicle with smallest additional workload* (FJWδ) policy. FJWπ is not optimal in light load and stabilizes the system only for a fraction of the possible arrival rates. The novel FJWδ is optimal in light load and stabilizes the system for almost all arrival rates. Fourth, we show that FJN and FJW exhibit a tipping point behavior: One vehicle makes the difference between almost optimal performance and instability. Therefore, careful dimensioning is required. In contrast to this, FJWδ can stabilize the system for almost all arrival rates, as long as the number of vehicles per depot is sufficient. This finding yields a connection between dimensioning and control of the system. Fifth, this last finding enables proper dimensioning of the delivery system controlled with FJWδ. In particular, the infrastructure of depots is subject of a long-term decision, the number of vehicles can be modified in short-term. The setup of the system (number of depots and

vehicles) shapes its financial costs and, in conjunction with the job assignment policy, the service quality provided to the customers for which they are willing to pay. Hence, which minimum expenditure is required to provide a certain service quality is an interesting question with respect to investing in such a delivery system. The lower bound mentioned in the first point provides the answer. We illustrate the application of this service-possibility-frontier with parameters reported by the company Matternet. Parts of this article related to the first, the second, and the fifth points are published in Grippa et al. (2017). The bound in the first point has been improved. All results and discussions related to FJW policies are novel.

2 Related work

2.1 Types of vehicle routing problems (VRPs)

The framework of the used model dates back to Dantzig and Ramser (1959), who introduced their formulation of the vehicle routing problem (VRP) as a generalization of the traveling salesman problem (Flood 1956). Since then, the operations research community has intensively studied how a central planner determines optimal sets of routes for fleets of homogeneous vehicles, supplying given sets of geographically dispersed customers with goods (Golden and Assad 1998). In the context of a "classical VRP", such an optimal set of routes accomplishes that (i) all customers are supplied with the requested products, (ii) none of the vehicles exceeds its capacity traveling along its route, (iii) no customer is visited more than once, (iv) all routes start and end at one central depot, and (v) the overall routing cost is minimized. In practical applications, VRPs have a broad diversity of additional requirements and operational constraints affecting the construction of the optimal set of routes. Among these are periodic VRPs (Angelelli and Speranza 2002), VRP with pickup and delivery (Desaulniers et al. 2002), VRP with split deliveries (Dror et al. 1994), and VRP with time windows to serve customers within (Cordeau et al. 2002). For reviews of exact and approximate methods for solving the classical VRP, we refer to Baldacci et al. (2007), Cordeau et al. (2007), Laporte (1992, 2007, 2009), Toth and Vigo (2002a, b) and, for an exhaustive bibliography on vehicle routing, to Laporte and Osman (1995).

VRPs are classified according to the nature of system input information. If all system input is known before the vehicles leave the depot(s) and does not change during mission execution, the matter of concern is described in the paragraph above, and said to be both static and deterministic. For many real-world applications at least some input information like customer arrivals behaves according to a probability distribu-

tion rather than is known a priori. These VRPs are denoted as stochastic. If some input information appears or changes during mission execution and has to be integrated immediately into decision-making, the VRP is called dynamic (Psaraftis 1998, 1995, or, for a recent review, Pillac et al. 2013). Then, designing sets of routes has to be replaced by designing routing policies which describe the evolution of motion paths as a function of newly arriving input.

2.2 Stochastic and dynamic vehicle routing in robotics and aeronautics

Bertsimas and Ryzin (1991, 1993a, b) were the first to comprehensively analyze a stochastic and dynamic VRP. The problem is also named dynamic traveling repair-person problem (DTRP): Requests are distributed according to a space-temporal stochastic process. Every request is associated with one location. Vehicles travel from request to request and spend time in each location providing on-site service. This problem received considerable attention for applications in robotics and aeronautics. Particular attention was devoted to the motion coordination of mobile robots, which includes spatially-distributed surveillance policies for UAVs that are adaptive to network changes (Bullo et al. 2011; Enright et al. 2009; Frazzoli and Bullo 2004; Pavone et al. 2011; Savla et al. 2008). In some of these policies, named partitioning policies, the service area is partitioned in sub-areas, one for each vehicle, and every vehicle serves only the requests in its sub-area according to certain rules (Pavone et al. 2011). Strategies were developed to ensure that a certain fraction of requests is served before the jobs expire (Pavone et al. 2009), that account for service priorities (Smith et al. 2010) and translating requests (Bopardikar et al. 2010), and that accomplish an effective system management without explicit communication (Arsie et al. 2009).

Considerably less work was dedicated to another vehicle routing problem called dynamic pickup and delivery problem (DPDP) (Swihart and Papastavrou 1999; Waisanen et al. 2008). In this problem, every request is associated with two locations (pickup and delivery) and, at each service, a vehicle transports a good from pickup to delivery location. This problem is different from the DTRP because the service of the request requires the vehicle to change location. Both pickup and delivery locations are drawn according to a continuous spatial distribution, i.e., goods are not stored in fixed depots.

In our problem, goods are available in depots with fixed locations. This problem can be seen either as a DTRP with a non-euclidean distance, or as a DPDP with pickup locations chosen from the depot locations. For these reasons, the approaches developed in related work are not directly applicable to this problem. Furthermore, we explicitly take battery charging into account.

3 System model

3.1 Entities of a delivery system

The system is composed of $K \in \mathbb{N}$ vehicles moving in a bounded and convex service area $\mathcal{A} \subset \mathbb{R}^2$ of size $A := \|\mathcal{A}\|$, where $\| \cdot \|$ is the Euclidean norm. A vehicle is denoted by v_k with identifier $k \in \{1, \ldots, K\}$. The current position of v_k at time t is $\mathbf{v}_k(t) \in \mathcal{A}$ with $t \geq 0$. All vehicles travel at the same constant velocity $\nu \in \mathbb{R}^+$ and are equipped with a battery, whose charging level at time t is represented by $b_k(t) \in [0, 1]$. The fact that batteries have to be recharged or exchanged is quantified by the parameter $\alpha \in (0, 1]$, which is the air-time ratio, i.e., $\alpha =$ air time/(air time + charge time).

The arrival of delivery requests for goods within $\mathcal{A}$ is generated by a Poisson process with finite intensity $\lambda \in \mathbb{R}^+$, where λ is the arrival rate. The requests, also called jobs, are indexed by the job identifier $n \in \mathbb{N}$, which indicates the order of request arrivals. The corresponding customer is called c_n; his or her position is denoted by $\mathbf{c}_n \in \mathcal{A}$ and assumed to be independently and uniformly distributed in $\mathcal{A}$. The information about the waiting customers is centrally stored. In distributed policies (NJR), vehicles periodically access the information and select the jobs. In centralized policies (FJN and FJW), a central unit periodically asks the vehicles for information and assigns the jobs to them. We assume that communication happens just before selection/assignment. However, in FJW policies, communication and assignment can be postponed without any effect on the policy outcome (see Sect. 6.1). Customer requests do not only differ with respect to timing and locations but also with respect to the goods requested to be delivered.

Goods are, in general, different but have the same expiration date and are treated with identical priority. The system consists of $L \in \mathbb{N}$ depots to store goods. The depots are interconnected and provide a sufficient number of all goods and service activities, like recharging batteries. We assume that, for capacity reasons, a vehicle cannot serve more than one customer request per trip (see Bertsimas and Ryzin 1993a, p. 71). The depots are set up at locations $\mathbf{d} = [\mathbf{d}_1 \ \mathbf{d}_2 \ \ldots \mathbf{d}_L] \in \mathcal{A}^L$, where $\mathbf{d}$ is chosen such that the expected distances between a random point $\mathbf{q} \in \mathcal{A}$ (potential request) generated according to a uniform distribution over $\mathcal{A}$ and the closest depot are minimal:

$$H_L(\mathbf{d}, \mathcal{A}) := \frac{1}{A} \cdot \int_{\mathcal{A}} \min_{l:l \in \{1,\ldots,L\}} \|\mathbf{d}_l - \mathbf{q}\| d\mathbf{q}. \tag{1}$$

This corresponds to the solution of the continuous multimedian problem known from geometric optimization (Papadimitriou 1981; Zemel 1984):

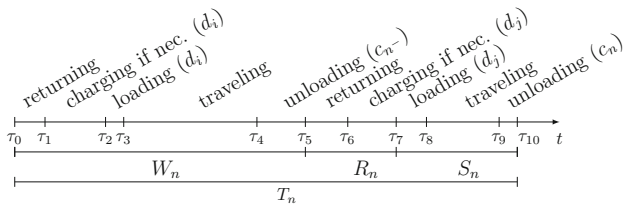$$\mathbf{d}^* = \arg \min_{\mathbf{d} \in \mathcal{A}^L} H_L(\mathbf{d}, \mathcal{A}), \tag{2}$$

Fig. 1 Time intervals involved in a customer service: delivery time T, waiting time W, return time R, and service time S

with $H_L^*(\mathcal{A}) := H_L(\mathbf{d}^*, \mathcal{A}).$ (3)

A depot is a storage but not a permanent home base for particular vehicles. Whenever a vehicle delivered a good, it either approaches the nearest depot or one that is more suitable to handle the next customer request. Suitability is determined by the job assignment policy.

3.2 Service operations and delivery time

The delivery time for customer c_n ($n \in \mathbb{N}$) is denoted by the stochastic variable $T_n = W_n + R_n + S_n$, where W_n is the waiting time, S_n is the service time, and R_n is the return time. Figure 1 illustrates all operations involved in the service of c_n from arrival at $t = \tau_0$ to service completion at $t = \tau_{10}$. The waiting time is $W_n = \tau_5 - \tau_0$. The return time is $R_n = \tau_7 - \tau_5$ and depends on the position of the customer c_{n^-} having been served before c_n and on the system's load. R_n is included in $[0, R'_n]$: It is null if the vehicle is ready at the depot, and maximum if the service of c_{n^-} is not completed at the arrival of c_n. The service time is $S_n = \tau_{10} - \tau_7$.

In the context of "classic" queuing theory, the term "service time" indicates the total time to process one request B_n, which corresponds to $S_n + R_n$ in this paper. Bertsimas and Ryzin (1991, 1993a, b) use service time to indicate the time spent on-site at the request location which corresponds to the term unloading time in this paper.

We neglect the times to load and unload goods. As introduced previously, the service provided to customers is the transport of goods from a depot to the customers. In practical applications it is reasonable to assume that loading and unloading times are negligible compared to the time needed for travel. These times could be modeled but besides slightly different numerical values they would not change the qualitative results of our analysis.

3.3 Queuing phenomena and stability

A job assignment policy Π has to restrain the outstanding jobs (Bertsimas and Ryzin 1991, 1993a, b; Bullo et al. 2011; Enright et al. 2009; Frazzoli and Bullo 2004; Pavone et al. 2011; Savla et al. 2008). Policy Π is referred to as stabilizing

if the expected number of pending jobs (customers waiting for service) stays confined over time, i.e., if there exists an arbitrary constant $\kappa < \infty$ such that

$$\bar{N}_\Pi := \lim_{t \to \infty} \mathbb{E}[N(t)|\Pi] \leq \kappa, \tag{4}$$

where $N(t)$ denotes the number of pending jobs at time t. We assume that $N(0) = 0$, i.e., no customer is waiting at $t = 0$.

The return and service times are crucial for the stability of the system. A necessary condition for stability is (Bertsimas and Ryzin 1993a, p. 63):

$$\frac{\bar{D}}{\nu} \leq \frac{K}{\lambda} \tag{5}$$

if the on-site service time is null. $\bar{D}$ is the average Euclidean distance between two customers served in sequence, λ is the arrival intensity, and ν is the vehicle speed. This condition applies to our problem with two changes: first, the distance becomes the distance function customer-depot-customer, which, divided by the speed, gives $\bar{R}' + \bar{S}$ (Fig. 1), where $\bar{R}'$ is the average return time in high load and steady state, and $\bar{S}$ is the average service time in steady state. Second, the effective number of used vehicles is on average αK, where α is the airtime ratio. Therefore, our stability condition is

$$\bar{R}' + \bar{S} \leq \frac{\alpha K}{\lambda} . \tag{6}$$

The problem analyzed in this paper can be modeled as an $M/G/K$ queue with interdependent service times B_n. M indicates Poisson distributed customer arrivals, G indicates service times distributed according to a generic distribution, and K is the number of servers. For $M/G/K$ queues with independent service times (Kleinrock 1975), the load factor is defined as

$$\rho := \frac{\lambda \bar{B}}{\alpha K} . \tag{7}$$

The system is said to be in low load condition if $\rho \to 0$ and in high load condition if $\rho \to 1$. A necessary condition for the stability of the system is $\rho < 1$. In case of stability, ρ can be interpreted as the expected value of the fraction of busy servers. This definition does not apply to our case because $\bar{B}$ depends on the system state (Kleinrock 1975), which includes the number of waiting customers. Nevertheless, the condition of stability is still valid: $\bar{B}$ approaches $\bar{R}' + \bar{S}$ for $\rho \to 1$, leading to (6). In words, to stabilize the system, it is necessary that the average time between two successfully completed service requests is not larger than the average time between two customer arrivals multiplied by the average number of available vehicles.

4 Expenditure for minimum infrastructure

We start to derive the minimum expenditure for infrastructure necessary to build a stable system as a function of system performance by deriving a lower bound on average delivery time $\bar{T}$. The average delivery time (with respect to the request location) is minimum if a vehicle is ready at the depot nearest to the request location. In this case the return time is null. Thus, the minimum average delivery time is equal to the minimum average service time: $\bar{T} \geq \bar{T}_{\min} = \bar{S}_{\min}$. The latter can be expressed in terms of the L-median function (3) as

$$\bar{S}_{\min} = \frac{H_L^*(\mathcal{A})}{\nu} . \tag{8}$$

Since the L-median function decreases with L, it is necessary to have a minimum number of depots L to be able to achieve the targeted performance $\bar{T}$. The minimum L as function of $\bar{T}$ is non-increasing piecewise constant, and defined as

$$L(\bar{T}) = \min \left\{ l \in \mathbb{N} : \frac{H_l^*(\mathcal{A})}{\nu} \leq \bar{T} \right\} . \tag{9}$$

In high load, the average time to process a job $\bar{B}$ is bounded according to $\bar{B} \geq 2\bar{S}_{\min}$. Using the latter in (7) together with $\rho < 1$, we derive a condition for the minimum number of vehicles necessary to have a stable system for a given configuration of depots:

$$K > \frac{\lambda 2 \bar{S}_{\min}}{\alpha} . \tag{10}$$

By using (8) in (10) and rounding to the nearest greater integer, we obtain

$$K(L) = \left\lceil \frac{2\lambda}{\alpha} \frac{H_L^*(\mathcal{A})}{\nu} \right\rceil . \tag{11}$$

For C_d and C_v denoting the costs of a depot and a vehicle, respectively, the total infrastructure expenditure is determined by the piecewise constant function

$$C(\bar{T}) = C_d L(\bar{T}) + C_v K(L(\bar{T})) . \tag{12}$$

For some combination of parameters, the function above increases with the targeted average delivery time $\bar{T}$: As $\bar{T}$ increases, the minimum number of depots $L(\bar{T})$ decreases, and more vehicles are needed to stabilize the system. Depending on the ratio between C_d and C_v this may increase the total infrastructure expenditure necessary to build a stable system, denoted by $I_{\min}$. Yet, $I_{\min}$ has to be a non-increasing function of $\bar{T}$. By construction, this is accomplished in the following way: For two configurations of depots with $L' < L''$, $\bar{T}'_{\min} > \bar{T}''_{\min}$, and $C' > C''$ we are able to obtain delivery

time $\bar{T}'_{\min}$ and reduce total expenditure by employing $L = L''$ instead of $L = L'$ depots alongside delaying any delivery by $\bar{T}'_{\min} - \bar{T}''_{\min}$ time units. This yields

$$I_{\min}(\bar{T}) = \min_{l:l \in \{L(\bar{T}),...,\infty\}} C_d \, l + C_v \left\lceil \frac{2\lambda}{\alpha} \frac{H_l^*(\mathcal{A})}{\nu} \right\rceil , \tag{13}$$

which constitutes the minimum infrastructure expenditure *necessary* for building a stable system that meets a targeted performance $\bar{T}$. Note that (13) is *sufficient* to guarantee system stability if the job allocation policy can stabilize the system for all $\rho < 1$.

The minimum infrastructure expenditure depends on the shape of the service area $\mathcal{A}$ through the terms $H_L^*(\mathcal{A})$ and $L(\bar{T})$. It is possible to obtain a lower bound to $I_{\min}(\bar{T})$ which does not depend the shape of the service area by considering

$$H_L^*(\mathcal{A}) \geq a\sqrt{\frac{A}{L}} , \tag{14}$$

where $a = 2/(3\sqrt{\pi})$, which is equivalent to considering circular sub-areas (Zemel 1984). If the right hand side of the equation above is used instead of the L-median function, the minimum number of depots is

$$L(\bar{T}) = \left\lceil \frac{a^2 A}{\nu^2 \bar{T}^2} \right\rceil . \tag{15}$$

By using (14) and (15) in (13) we obtain the lower bound. In general, the bound is not tight but, depending on the parameters, can be very close to the actual minimum infrastructure.

We show in the following sections that one vehicle can make the difference between almost optimal performance and an unstable system (see Fig. 3). Therefore, choosing a sufficient number of vehicles and depots is crucial for a reliable system. We call this operation system dimensioning. In Sect. 7 we use (13) in a numerical example and discuss the result.

5 Simple job assignment policies

5.1 Description of policies

Job assignment goes beyond merely picking the next request. It has to specify all decisions needed to operate a delivery system including which customer request to serve first, which vehicle to let serve the next customer request, at which depot to let vehicles load up goods, when to recharge the battery, and where to let vehicles return to if no customers are waiting. We first analyze two simple classes of job assignment policies: nearest job first to random vehicle (NJR) and first job first to nearest vehicle (FJN). NJR policies select jobs

Table 1 Features of job assignment policies

Feature	Feature details	NJR soon	NJR late	FJN soon	FJN late
Order of selection	Jobs are selected in FCFS order from a shared queue	–	–	✓	✓
	There is no specific assignment order of jobs	✓	✓	–	–
Decision location	Decisions at the customer, just after completing a service	✓	–	✓	–
	Decisions at the depot, just before loading goods	–	✓	–	✓
Coordination mechanisms	If many vehicles seek to select a job, a random one does	✓	✓	–	–
	If many vehicles seek to select a job, the nearest one does	–	–	✓	✓
	If there are no jobs a vehicle goes to the nearest depot	✓	✓	✓	✓
Battery management	If a vehicle's battery level is below 30% the vehicle approaches the nearest depot. If a vehicle's battery reaches a charging level higher than 80% the vehicle is ready for service. If there is no customer the vehicle fully recharges its battery	✓	✓	✓	✓

based on the location of the customer; FJN policies select jobs based on the arrival time of the customer requests. By comparing these two policies, we seek to gain insights into the following question: Is it worth to delay the decision on job assignment to obtain more information about new customer requests in order to reduce the delivery time? For both classes we evaluate two extreme cases: when assignment is made as soon as possible (just after the previous service, NJR-soon and FJN-soon), and when assignment is made as late as possible (just before loading the good, NJR-late and FJN-late).

A first issue is *the timing of job assignments*. It seems plausible to assume that decisions should be made as late as possible to utilize the most recent information about the system status, basically postponing the individual job assignments until loading goods in the depot. Such delayed decisions can also have a negative effect. This relates to the fact that, for $L \geq 2$, one of the most important decisions to make is where a vehicle should return to after satisfying a customer request. If a non-postponed decision about the next job includes a "clever" assignment of the depot to travel to, while a postponed decision leads to being at a suboptimal place at the moment of job assignment, the advantage gained by processing more recent information is counteracted by the disadvantage of the extra distance traveled to the customer to be served next. We will investigate this effect in Sect. 5.4.

A second issue is the *coordination mechanism* to avoid that more than one vehicle is assigned to the same job. Such a scenario, where a particular customer, say $c_{\hat{n}}$, is selected by K' vehicles with $2 \leq K' \leq K$, can easily happen if the system is not fully utilized, i.e., for $\rho \to 0$. FJN policies are centralized: A central entity selects the next job and assigns it to the nearest vehicle. Hence, the assignment is based on arrival time and localization of waiting customers, positions of depots and vehicles. NJR policies are distributed: Every vehicle selects jobs based on its position with respect to the position of customers and depots. Whenever a vehicle selects

a job, it is removed from the list of waiting jobs. Every vehicle knows its own position, the positions of depots and of waiting customers. For these policies, the coordination mechanism is randomized: If more than one vehicle is ready to take a new job at a given time, a random priority order is assigned to the vehicles involved. The randomize mechanism corresponds to a real case where vehicles are not coordinated at all and the connection delay to the list of waiting customers is not predictable because it depends on the network conditions.

Table 1 shows all simple policies and classifies them according to three features: the order of jobs to be done, the timing of assignment, and the coordination of jobs in case of ambiguous assignments. A feature included in all policies is the effect of a vehicle v_k's battery level, $b_k(t)$, on the selection of new jobs. In particular, if at the time of job selection/assignment the battery level is below 30%, the vehicle approaches the nearest depot to recharge and does not select a job until its battery level has reached 80%. If no service request arrives, v_k fully recharges.

5.1.1 NJR policies

In NJR-soon—after completing a service—the vehicle v_k selects the next customer such that the travel distance to this customer via a generic depot is minimized. In mathematical terms we solve

$$\min_{\substack{l:l \in \{1,\ldots,L\} \\ n:c_n \in \mathcal{N}(\tau)}} \|\mathbf{v}_k(\tau) - \mathbf{d}_l\| + \|\mathbf{d}_l - \mathbf{c}_n\|, \tag{16}$$

where τ is the time instant of completion of the previous job, which is here equal to the time instant of the new job assignment. Such assignment is made if the battery level at time τ is high enough, i.e., $b_k(\tau) \geq 0.3$; otherwise a vehicle will return to the nearest depot. Every vehicle has to make $L \cdot N(\tau)$ comparisons to conclude an individual job assignment. If a job is at minimum traveling range for more than

one vehicle, according to the random coordination described above, only one randomly chosen vehicle will serve the job.

In NJR-late, after completion of a service, the vehicle v_k first travels to the closest depot. The vehicle then selects the nearest customer, i.e., it solves:

$$\min_{n:c_n \in \mathcal{N}(\tau+R)} \|\mathbf{v}_k(\tau+R) - \mathbf{c}_n\|, \tag{17}$$

where $\tau + R$ is the time instant v_k arrives at the depot. The vehicle remains in the depot if there is no waiting customer. If a job is at minimum traveling range for more than one vehicle, the supplying vehicle is determined randomly. This policy requires $L + N(\tau + R)$ comparisons.

5.1.2 FJN policies

Following a first come first served (FCFS) policy, a central unit always selects the "oldest" unaddressed request, i.e., the smallest n with $c_n \in \mathcal{N}(t)$, where t indicates the instance of the job assignment. Let this request be called $\underline{n}(t)$. In particular, policy FJN-soon implies that whenever at least one vehicle completes a service at time $t = \tau$ a central controller selects the next customer in line, i.e., request $\underline{n}(\tau)$, and chooses the vehicle v_{k*} and the path through the depot, d_{l*}, that minimizes the total distance between its current position $\mathbf{v}_k(\tau)$, a depot's position $\mathbf{d}_l$, and the customer's position $\mathbf{c}_{\underline{n}(\tau)}$:

$$\min_{\substack{l:l \in \{1,...,L\} \\ k:k \in \mathcal{K}'}} \|\mathbf{v}_k(\tau) - \mathbf{d}_l\| + \|\mathbf{d}_l - \mathbf{c}_{\underline{n}(\tau)}\|, \tag{18}$$

where $\mathcal{K}'$ is the number of vehicles ready for the service. Obeying (18), a total of $K' \cdot L$ computations is needed to come up with a vehicle's individual assignment of whom to serve next.

Obeying the rules of policy FJN-late, whenever at least one vehicle is available at a depot, the central unit selects the request $\underline{n}(\tau+R)$ and performs K' computations to determine the vehicle nearest to that job. Subsequently, L computations are performed to find the depot closest to the selected customer where the vehicle has to return to after delivery:

$$\min_{k:k \in \mathcal{K}'} \|\mathbf{v}_k(\tau+R) - \mathbf{c}_{\underline{n}(\tau+R)}\| + \min_{l:l \in \{1,...,L\}} \|\mathbf{c}_{\underline{n}(\tau+R)} - \mathbf{d}_l\|. \tag{19}$$

5.2 Simulation setup

For each of the simple policies, we simulate the movement of $K \in \{1, \ldots, 24\}$ vehicles in a square area of $A = 16$ km^2. The $L \in \{1, 4, 9, 16\}$ depots are located such that the average distance of any potential request in $\mathcal{A}$ from the nearest depot is minimized. Customer requests are assumed to randomly arrive over time at a constant rate $\lambda = 0.65$ requests/min. All vehicles travel at a constant velocity of $v = 30$ km/h, neglecting acceleration and deceleration phases and neglecting the extra time for starting and landing. The vehicles' air time is limited due to their finite battery capacity, which is assumed to be full at $t = 0$. We choose the ratio between air time and charge time as being $1/3$, i.e., $\alpha = 0.25$. Moreover, every v_k is placed at one of the depots at $t = 0$. Finally, we neglect the loading time of goods.

In the simulations, the system presented in Sect. 3 is observed every δ time units. Interarrival times are generated according to a geometric distribution so that the number of customer arrivals per time unit follows a binomial distribution with parameters $h = 1/\delta$ and $p = \lambda\delta$. The Poisson distribution is a sufficient approximation of the binomial distribution if $p \leq 0.08$ and $h \geq 1500p$ (Bronshtein et al. 2007). Therefore, for all of our simulation runs, we choose δ such that $\delta \leq \min\{0.08/\lambda, 1/\sqrt{1500\lambda}\}$.

The data collected from the simulations are used to compute the average delivery time. Specifically, we firstly estimate the length of the warm-up phase using Welch's method (1983). Secondly, we compute the average delivery time with the replication/deletion method (Law 2007), which is equivalent to the independent replications method (Welch 1983). These methods are used for statistical analysis of data and enable us to estimate expected values and confidence intervals.

For every parameter setup, we perform 10 simulations with a minimum of 4000 jobs for every simulation. If it is impossible to evaluate whether the system reaches the steady state within this period, we simulate 10,000 customers requests which is more than 10 days of continuous operation.

5.3 Warm-up phase

We discuss the computation of the warm-up phase for NJR-soon and FJR-soon for $K \in \{11, 12, 14, 16\}$ vehicles and $L = 16$ depots. Figure 2 reveals that the length of the transient phase decreases with an increasing number of vehicles, i.e., with a decreasing load factor. We conclude that delivery systems provided with fewer vehicles must be simulated for a longer period (higher number of demands $\tilde{n}$) to reach steady state conditions. Being conservative, we estimate that the length of the warm-up phase is $\tilde{n}_{wu} = 3000$ for the case $K = 11$, while being $\tilde{n}_{wu} = 2000$ for the case $K = 12$, and $\tilde{n}_{wu} = 500$ for all other cases.

Moreover, Fig. 2 introduces a result addressed when discussing Fig. 3d below: FJN-soon can yield a substantially better system performance than NJR-soon. However, FJN-soon cannot stabilize the system for $K = 11$.
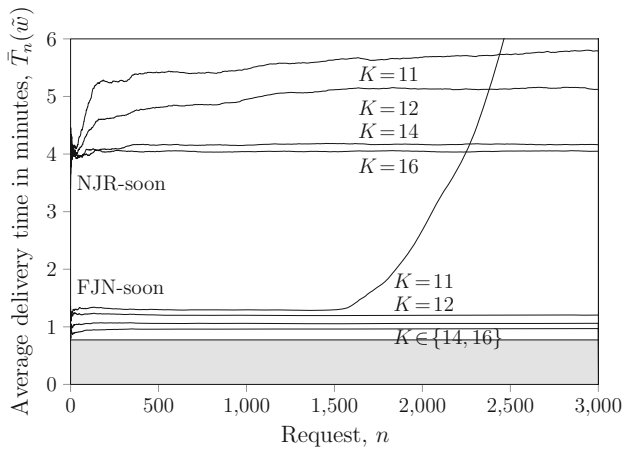
Fig. 2 Delivery time averaged over 10 simulation runs and $2\tilde{w} + 1 = 1001$ demands vs. demand index n for K vehicles; $L = 16$, $A = 16$ km^2, $\lambda = 0.65$ req./min., $v = 30$ km/h, and $\alpha = 0.25$

5.4 Performance evaluation and lesson learned

Figure 3 shows the average delivery time $\bar{T}$ for different job assignment policies as a function of the number of vehicles K and the number of depots L. The shaded areas indicate "impossible regions", i.e., all ordered pairs $(K, \bar{T})$ for which either $\rho > 1$ or $\bar{T} < \bar{T}_{\min}$ or both. The following basic phenomena can be observed.

5.4.1 Coordination mechanism and low load

FJN policies are optimal in light load, i.e., the best vehicle to serve a request (in order to minimize the delivery time) is the vehicle nearest to that request. If the system is in light load, increasing the number of vehicles K has no effect on the performance. Additional vehicles are not used because, on average, there is a vehicle ready to serve every new request from the nearest depot. As the load factor increases, FJN policies become unstable for arrival rates λ rather far from the
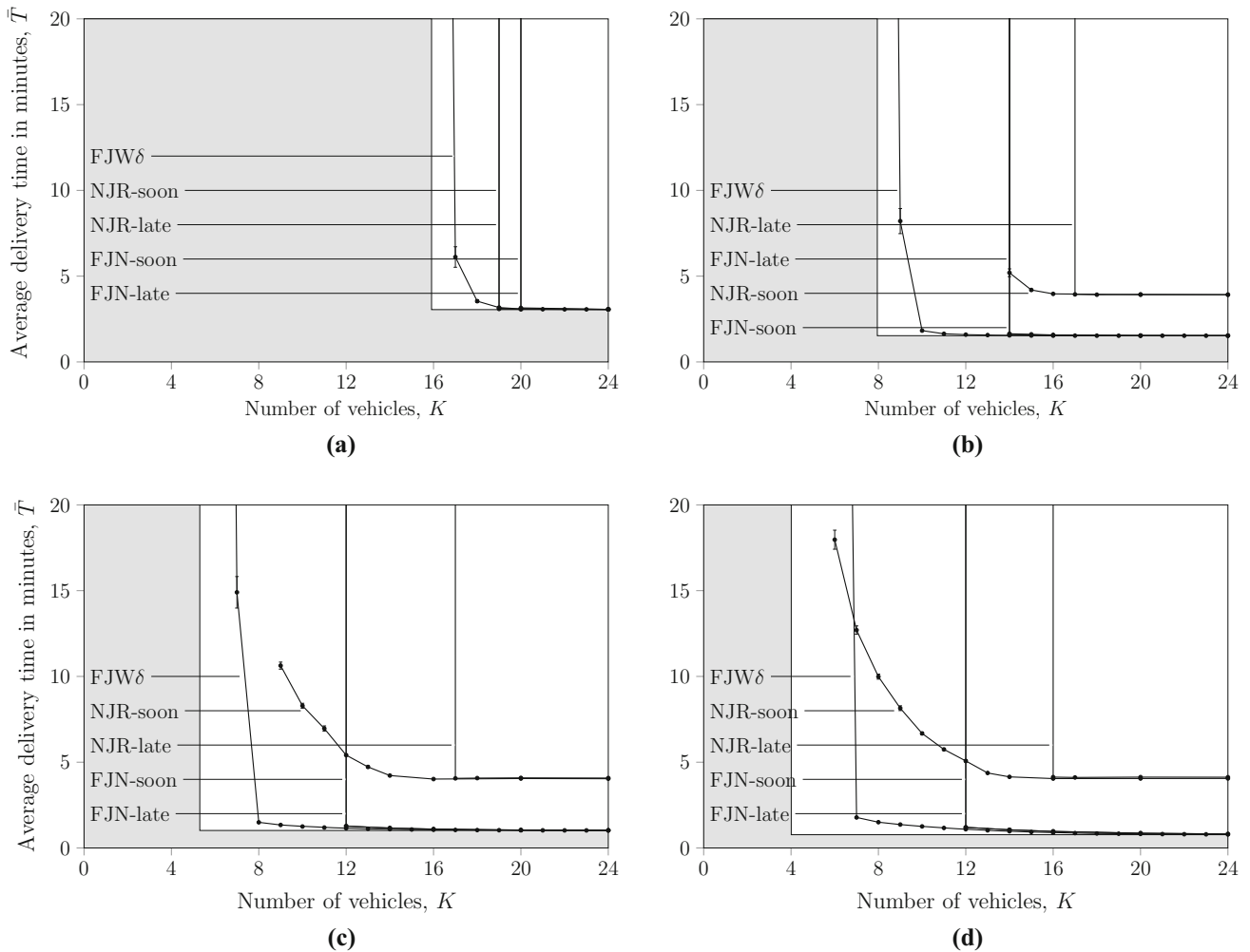

Fig. 3 Expected delivery time (90% confidence) versus number of vehicles for L depots; $A = 16$ km^2, $\lambda = 0.65$ req./min., $v = 30$ km/h, and $\alpha = 0.25$. **a** Number of depots $L = 1$. **b** Number of depots $L = 4$. **c** Number of depots $L = 9$. **d** Number of depots $L = 16$

theoretical limit. This is probably due to the non-optimal battery management. In high load, vehicles spend a significant amount of time recharging the battery while the customers are waiting. Therefore, a good policy has to include the battery level in the job allocation decision.

The randomized coordination mechanism of NJR policies performs poorly in light load. Even if there are vehicles available a the depot nearest to a new request, the serving vehicle might come from another depot. This increases the minimum average service time and dramatically degrades performance.

5.4.2 Tipping point behavior

As the number of vehicles decreases, FJN policies show a tipping point behavior: One vehicle makes the difference between almost optimal performance and instability. In FJN policies, as the number of vehicles decreases beyond a certain threshold, the vehicles often have to change depot. Such depot switching increases the minimum service time making the system unstable. The phenomenon is made worse by the non-optimal battery management. In terms of queuing theory, this extreme behavior is explained by the fact that the delivery time grows much faster with the parameters defining the load factor (λ, $\bar{B}$, α, K) than in "classic" queues. Bertsimas and van Ryzin report the same for waiting time and arrival rate (Bertsimas and Ryzin 1991). In a "classic" queue, a decrement in the number of vehicles K would increase the load factor according to $\lambda\bar{B}/(\alpha(K - \Delta K))$. In our problem, the average time to process a request $\bar{B}$ increases as the number of vehicles decreases beyond a certain threshold. Therefore, the load factor increases according to $\lambda(\bar{B} + \Delta B)/(\alpha(K - \Delta K))$, which determines higher increment on the delivery time. Given this relationship between delivery time and number of vehicles and the fact that the number of vehicles is discrete, one vehicle can make the difference between being in mid/light load and exceeding the stability region. The difference in stability of FJN and NJR policies derives from the differing ability to keep $\bar{B}$ small as the load factor increases.

Despite the random coordination, NJR-soon performs better than the FJN policies in high load. In high load, there are fewer conflicts, and assignment of the nearest job limits the transitions between depot sub-areas (with high probability there are requests in the same sub-area). However, this capacity to stabilize the system for higher arrival rates comes at the price of lack of fairness. Requests are treated with priorities depending on the distance from the depot. Therefore, there is a set of demands that is served with long delivery time.

5.4.3 Timing matters

Using FJN policies, timing has no effect on delivery time and system stability (see Fig. 3). In contrast, using NJR policies,

an early decision is beneficial for stability, i.e., NJR-soon is more robust than NJR-late. The positive effect of more recent information outweighs the negative effect of returning to a depot located sub-optimally (on average) for the next job. The gap between the two NJR policies widens with more depots, i.e., the effect of "being at a suboptimal place" is more pronounced in systems with many depots. This happens because in heavy load conditions randomized job coordination is rarely necessary (if at all), and NJR behaves similar to a nearest neighbor policy (as mentioned above) which serves the nearest demand after every service completion.

6 Workload-based job assignment

6.1 Description of policies

We now study two policies that take into account the workload of vehicles. The first is known from queuing theory (Asmussen 2003); the second is an extension based on the lessons learned from the analysis of simple policies. The workload at time t is the remaining time that a vehicle needs to complete all assigned jobs.

The first policy is *first job to the vehicle with the smallest workload* (FJWπ). It serves jobs in an FCFS order and assigns them to the vehicle with the smallest workload. At every job arrival, a central unit asks each vehicle to compute its current workload. All vehicles send back their computed values and the central unit assigns the job to the vehicle with the minimum value. A vehicle with no workload heads toward the nearest depot to charge its battery. For a standard $G/G/K$ queue, this policy minimizes all moments of the waiting time, number of jobs in the system (queue length), and workloads for all vehicles (Asmussen 2003). In our delivery system, however, the policy performs poorly because it does neither consider the positions of the vehicles relative to the customer and depots nor the energy needed to do the job.

The second policy is *first job to the vehicle with the smallest additional workload* (FJWδ). Instead of using the current workload of a vehicle as an assignment criterion, this policy employs the amount of workload that the new job adds to the current workload. The key idea behind this policy is to minimize the overall workload in the entire system in the long run. FJWδ considerably increases stability and reduces delivery time. The additional workload takes into account the way from the previous customer to the depot in order to pick up the new good (or the current location if the vehicle is not serving), battery charging in the depot (to reach the new customer and come back to the nearest depot), and transport the good from the depot to the customer. If a vehicle is currently not serving a customer, the current location is taken instead of the customer location.

It should be emphasized that the policy is scalable with the number of vehicles and depots: Each vehicle computes the minimum of L additional workload values to choose the depot for every assignment, and the central unit computes the minimum of K additional workload values to choose the vehicle. The policy can thus be applied in large systems. Alternatively, it can be applied locally by dividing the multidepot system in many single-depot systems.

Some design choices of FJWδ can be traced back to the lessons learned from the simple policies: First, we focus on FCFS to have fairness among requests with varying distances from the nearest depot. Second, we explicitly take into account the battery level in job assignments. Third, assignments can be postponed without impact on performance until the workload of the chosen vehicle is zero. This is important from an implementation point of view as the system can be well operated in areas with poor network coverage. Vehicles can exchange information with the central unit whenever they are intermittently connected. Note that good assignments are especially important in systems with high load; since vehicles in high load have on average a long time to reach a workload of zero, the communication can in fact be postponed over a longer period. In this way, FJWδ combines the best of two NJR policies: NJR-soon has good performance but strict connectivity requirements; NJR-late has relaxed connectivity requirements but bad performance.

6.2 Performance evaluation

Figure 4 shows the expected delivery time over the job arrival rate for FJW policies. FJWπ is suboptimal in low load (similar to NJR) and cannot stabilize the system for high load. This is because the policy assigns the job to the vehicle that will be ready for service first but which might be far away from the customer. In contrast, FJWδ is optimal in low load
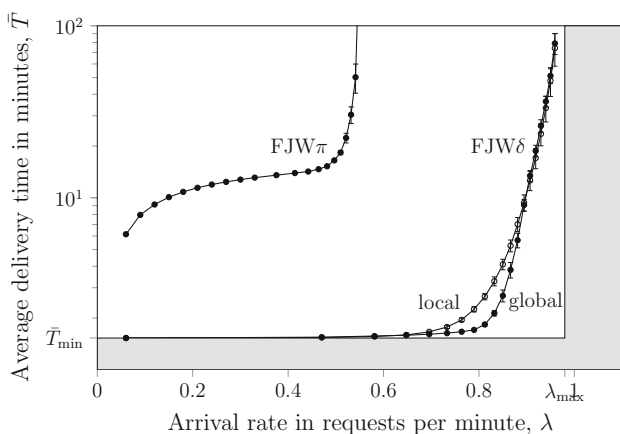

Fig. 4 Expected delivery time (90% confidence) for workload-based job assignment policies; $L = 4$, $K = 12$, $A = 16$ km^2, $v = 30$ km/h, and $\alpha = 0.25$

and performs very well in high load. It is optimal in low load because the chosen vehicle typically comes directly from the depot nearest to the customer it will serve.

A necessary condition to have an optimal policy is that it performs at least as good on a global level as on the local level. This is true in our case: FJWδ applied at local level performs worse than at global level for mid load. The reason is as follows: In mid load, there is a benefit in exchanging vehicles between different depots, so that vehicles from depots with few jobs can help in depots with many jobs. As the load increases, all depots have many jobs, thus vehicles tend to stay at their depot.

It is of interest for dimensioning and controlling the system to investigate whether FJWδ can stabilize the system for all possible load factors ρ, defined by starting from (7) considering $\bar{B} \geq 2\bar{S}_{\min}$ (in high load) and using (8):

$$\rho = \frac{\lambda}{\alpha K} \frac{2H_L^*(\mathcal{A})}{v} . \tag{20}$$

Simulation results suggest that this is true if the number of vehicles per depot K/L is large enough. Figure 5 shows the maximum load factor for which the policy is able to keep the system stable:

$$\rho_{\text{stable}} = \sup_{0 \leq \rho \leq 1} \left\{ \lim_{t \to \infty} \mathbb{E}[N(t)|\rho, \text{FJW}\delta] \leq \kappa \right\} \tag{21}$$

with some $\kappa < \infty$. Let us interpret the results in more detail. If there are many vehicles per depot, the average return plus service time in high load depends on the distance between customer and nearest depot, i.e., is $2H_L^*(\mathcal{A})/v$. With decreasing number of vehicles, to serve customers in an FCFS order, vehicles do not stay in the same depot area but change depots. Once this occurs, the average return time increases and the system soon becomes unstable. Curve fitting suggests that the value of the stability threshold ρ_{stable} increases with K/L according to an exponential law:

$$\rho_{\text{stable}} = 1 - e^{-\gamma K/L} \tag{22}$$

with $\gamma = 2.73$ in this case. Due to this law, only a few vehicles per depot are needed to stabilize the system for almost all load factors. For example, a load of $\rho_{\text{stable}} = 90\%$ can be carried with only one vehicle per depot.

Results of Fig. 5 are obtained as follows: For every given combination of (L, K), we find the stability threshold ρ_{stable} by visual inspection of $\bar{T}(\lambda)$-plots. Each point on the $\bar{T}(\lambda)$-plot is obtained by simulating the system ten times over 100,000 jobs. In all scenarios, depots are located to minimize the L-median function.

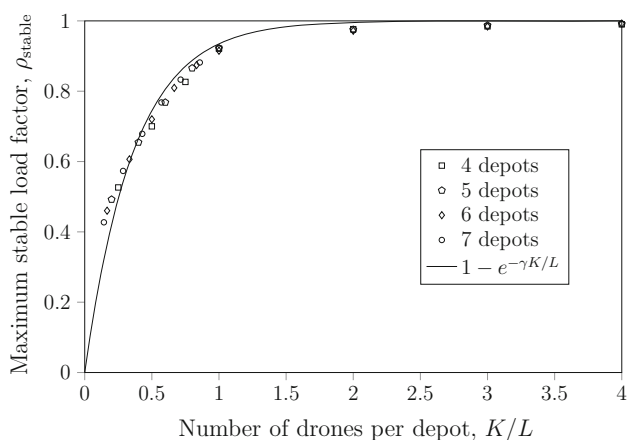A comparison of FJWδ with the simple policies NJR and FJN confirms these findings (see Fig. 3). FJWδ outperforms

Fig. 5 Maximum load factor reachable by the policy FJWδ as function of the number of vehicles per depot; $A = 16$ km^2, $v = 30$ km/h, and $\alpha = 0.25$

both in all cases except in the configuration $(K, L) = (7, 16)$, where NJR-soon seems to have a lower average delivery time. FJWδ is still preferable because of fairness issues with NJR-soon. Even though FJWδ works very well, when looking at the system from the dimensioning point of view, the tipping point behavior cannot be eliminated. Therefore, a reliable system inevitably calls for careful dimensioning.

7 Dimensioning

For this reason we propose a method for the systems dimensioning: selection of the number of vehicles and depots. We have seen that the number of depots must be chosen in relation to the size of the service area, and that this long-term choice on depot infrastructure must be coordinated with the short-term choice on vehicles. For company-specific parameter values, a diagram like Fig. 6 translates the insights derived in the subsection above into the monetary domain. It relates a company's expenditure I for depots and vehicles to average delivery time $\bar{T}$. The purpose of such a plot is to provide decision making support for companies that set up an airborne delivery system equipped with small UAVs.

To give an example, Fig. 6 is produced on the assumption that the cost of a UAV suitable to deliver two-kilogram packages is 1000 US$ plus a maintenance cost of 100 US$ per annum, and the cost of a depot is 15,000 US$ plus a maintenance cost of 500 US$ per annum. Operating the system over ten years, the costs per vehicle and depots are $C_v = 2000$ US$ and $C_d = 20,000$ US$, respectively. These parameter values are those assumed by the company Matternet (Raptopoulos 2012).

A lower bound on the expenditure required to build a stable system, denoted by I_{min}, is derived in Sect. 4 and is given by (13). Figure 6 plots this bound for the given parameters,
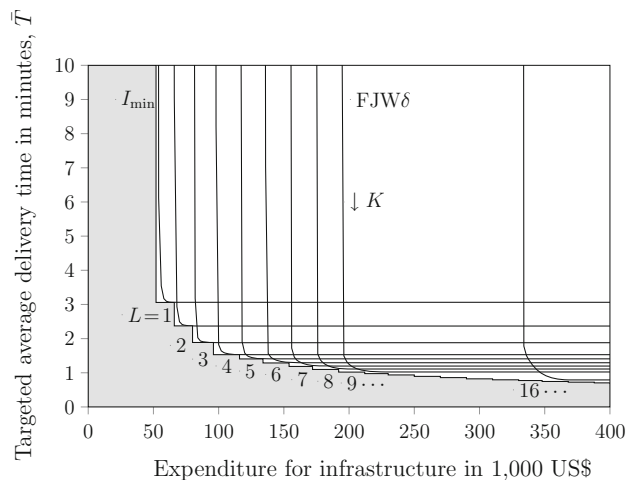

Fig. 6 Relation between infrastructure expenditure and average delivery time; $A = 16$ km^2, $\lambda = 0.65$ req./min., $v = 30$ km/h, $\alpha = 0.25$, $C_v = 2000$ US$, $C_d = 20,000$ US$

where all values below the bound are shown as a shaded area. The bound has a "staircase" shape with tread levels being the minimum average delivery time achievable with a particular number of depots. No operable system exists for parameters in this area, while every combination of I and targeted $\bar{T}$ located above fulfills $\rho < 1$ and $\bar{T} > \bar{T}_{min}$. The bound corresponds to a *service possibility frontier*; it gives a necessary but not sufficient condition for infrastructure expenditure. The actual performance is policy dependent, and more financial resources than I_{min} may be needed to operate the system in a stable manner and to meet the targeted performance. The performance of the FJWδ policy is plotted for scenarios with a varying number of depots.

A company that wants to operate a delivery service and serve a customer within a certain average delivery time can employ such a diagram as follows: If the average delivery time should be no more than τ, the company has to look for feasible combinations of infrastructure and stabilizing policies, i.e., squares and triangles, that are located as close as possible to the origin and below the $(\bar{T} = \tau)$-line. If the customers' willingness to pay for several levels of service quality is given, it is possible to quantify the company's marginal revenues of increasing performance. From Fig. 6 we know the marginal cost of decreasing delivery time. Then we are able to determine the infrastructure that maximizes the company's profit. For the parameters of Matternet we find that a system with $L = 1$ depot can serve a customer in about 3 min on average in an area of 16 km^2, which comes at a cost of 60,000 US$. This time can be reduced to less than 1.5 min if the company spends more than 100,000 US$ for infrastructure (associated with $L = 4$ depots). If the company's marginal revenue of reducing delivery time by 1.5 min (50%) is larger than approximately 40,000 US$, the four-depot configuration is better than the one-depot configuration. In other words,

Figure 6 informs about the financial resources required for achieving a certain quality of service with a certain policy and about the volume of additional financial resources obligatory to "buy" a shorter delivery time.

8 Conclusions

This article addresses the high-level control and dimensioning of a drone-based delivery system using simulations and queuing theory. It was found that job assignment policies can experience a tipping point behavior: A stable system could immediately become unstable if one vehicle fails. An advanced job assignment policy is proposed that uses the increment in workload as assignment metric: The job is assigned to the vehicle that will do the job faster than other vehicles. This policy, called *first job first to vehicle with the smallest additional workload*, leads to an optimal average delivery time for low loads and works very well up to high loads. It is scalable with the number of depots and vehicles. Simulation results indicate that the policy stabilizes the system for all loads if the number of vehicles per depot is sufficient. To account for the tipping point behavior we show how to dimension a stable delivery system for resolving the trade-off between expenditure and service quality. The dimensioning considers two time horizons: long-term decisions on the number of depots to deploy in the service area and short-term decisions on the number of vehicles to use. Future work will analyze systems with inhomogeneous customers and real-world data.

References

Andre, T., Hummel, K. A., Schoellig, A. P., Yanmaz, E., Asadpour, M., Bettstetter, C., et al. (2014). Application-driven design of aerial communication networks. *IEEE Communications Magazine*, *52*(5), 129–137.

Angelelli, E., & Speranza, M. G. (2002). A periodic vehicle routing problem with intermediate facilities. *European Journal of Operational Research*, *137*, 233–247.

Arsie, A., Savla, K., & Frazzoli, E. (2009). Efficient routing algorithms for multiple vehicles with no explicit communication. *IEEE Transactions on Automatic Control*, *54*(10), 2302–2317.

Asmussen, S. (2003). *Applied probability and queues* (2nd ed.). Berlin: Springer.

Baldacci, R., Toth, P., & Vigo, D. (2007). Recent advances in vehicle routing exact algorithms. *4OR*, *5*(4), 269–298.

Bertsimas, D. J., & van Ryzin, G. J. (1991). A stochastic and dynamic vehicle routing problem in the Euclidean plane. *Operations Research*, *39*(4), 601–615.

Bertsimas, D. J., & van Ryzin, G. J. (1993a). Stochastic and dynamic vehicle routing in the Euclidean plane with multiple capacitated vehicles. *Operations Research*, *41*(1), 60–76.

Bertsimas, D. J., & van Ryzin, G. J. (1993b). Stochastic and dynamic vehicle routing with general demand and interarrival time distributions. *Advances in Applied Probability*, *25*, 947–978.

Bopardikar, S. D., Smith, S. L., Bullo, F., & JaP, Hespanha. (2010). Dynamic vehicle routing for translating demands: Stability analysis and receding-horizon policies. *IEEE Transactions on Automatic Control*, *55*(11), 2554–2569.

Bronshtein, I. N., Semendyayev, K. A., Musiol, G., & Mühlig, H. (2007). *Handbook of mathematics* (5th ed.). Berlin, DE: Springer.

Bullo, F., Frazzoli, E., Pavone, M., Savla, K., & Smith, S. L. (2011). Dynamic vehicle routing for robotic systems. *Proceedings of the IEEE*, *99*(9), 1482–1504.

Cordeau, J. F., Desaulniers, G., Desrosiers, J., Solomon, M. M., & Soumis, F. (2002). The VRP with time windows. In *The vehicle routing problem, society for industrial and applied mathematics*, Philadelphia, PA, Chapter 7, pp. 157–172.

Cordeau, J. F., Laporte, G., Savelsbergh, M. W., & Vigo, D. (2007). Vehicle routing. In C. Barnhart & G. Laporte (Eds.), *Handbooks in operations research and management science* (1st ed., Vol. 14, pp. 367–428). Amsterdam, NL: Elsevier.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, *6*(1), 81–90.

Desaulniers, G., Desrosiers, J., Erdmann, A., Solomon, M. M., & Soumis, F. (2002). The VRP with pickup and delivery. In *The vehicle routing problem, society for industrial and applied mathematics*, Philadelphia, PA, Chapter 9, pp. 225–242.

Dror, M., Laporte, G., & Trudeau, P. (1994). Vehicle routing with split deliveries. *Discrete Applied Mathematics*, *50*(3), 239–254.

Enright, J. J., Savla, K., Frazzoli, E., & Bullo, F. (2009). Stochastic and dynamic routing problems for multiple uninhabitated aerial vehicles. *Journal of Guidance, Control, and Dynamics*, *32*(4), 1152–1168.

Flood, M. M. (1956). The traveling-salesman problem. *Operations Research*, *4*(1), 61–75.

Frazzoli, E., & Bullo, F. (2004). Decentralized algorithms for vehicle routing in a stochastic time-varying environment. In *Proceedings of IEEE conference on decision and control*, Paradise Island, pp. 3357–3363.

Golden, B. L., & Assad, A. A. (Eds.). (1988). *Vehicle routing: Methods and studies*. Amsterdam: Elsevier.

Grippa, P. (2016). Decision making in a UAV-based delivery system with impatient customers. In *Proceedings IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Daejeon, Korea, pp. 5034–5039.

Grippa, P., Behrens, D., Bettstetter, C., & Wall, F. (2017) Job selection in a network of autonomous UAVs for delivery of goods. In *Proceedings of robotics: Science and systems*, Cambridge, MA.

Kleinrock, L. (1975). *Queueing systems: Theory* (Vol. 1). New York: Wiley Interscience.

Kovacina, M. A., Palmer, D., Yang, G., & Vaidyanathan, R. (2002). Multi-agent control algorithms for chemical cloud detection and mapping using unmanned air vehicles. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Lausanne, CH, pp. 2782–2788.

Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, *52*(3), 345–358.

Laporte, G. (2007). What you should know about the vehicle routing problem. *Naval Research Logistics*, *54*(8), 811–819.

Laporte, G. (2009). Fifty years of vehicle routing. *Transport Science*, *43*(4), 408–416.

Laporte, G., & Osman, I. H. (1995). Routing problems: A bibliography. *Annals of Operations Research*, *61*(1), 227–262.

Law, A. M. (2007). *Simulation modeling and analysis* (4th ed.). Boston, MA: McGraw-Hill.

Lima, P., Floreano, D., Schill, F.S., & Basiri, M. (2014) Audio-based localization system for swarms of micro air vehicles. In *Proceedings of IEEE international conference on robotics and automation (ICRA)*, Hong Kong, CHN.

Papadimitriou, C. H. (1981). Worst-case and probabilistic analysis of a geometric location problem. *SIAM Journal on Computing*, *10*(3), 542–557.

Pavone, M., Bisnik, N., Frazzoli, E., & Isler, V. (2009). A stochastic and dynamic vehicle routing problem with time windows and customer impatience. *Mobile Networks and Applications*, *14*(3), 350–364.

Pavone, M., Frazzoli, E., & Bullo, F. (2011). Adaptive and distributed algorithms for vehicle routing in a stochastic and dynamic environment. *IEEE Transactions on Automatic Control*, *56*(6), 1259–1274.

Pillac, V., Gendreau, M., Guéret, C., & Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research*, *225*(1), 1–11.

Psaraftis, H. N. (1988). Dynamic vehicle routing problems. In *Vehicle routing: Methods and studies*, Elsevier, Amsterdam, pp. 223–248.

Psaraftis, H. N. (1995). Dynamic vehicle routing: Status and prospects. *Annals of Operations Research*, *61*(1), 143–164.

Quaritsch, M., Stojanovski, E., Bettstetter, C., Friedrich, G., Hellwagner, H., Rinner, B., Hofbaur, M., & Shah, M. (2008). Collaborative microdrones: Applications and research challenges. In *Proceedings of international conference on autonomic computer and communication systems*, Turin, IT, pp. 38:1–38:7.

Raptopoulos, A. (2012). Physical transport. Talk at Solve for X. https://youtu.be/7B3E4OOuGDk.

Savla, K., Frazzoli, E., & Bullo, F. (2008). Traveling salesperson problems for the dubins vehicle. *IEEE Transactions on Automatic Control*, *53*(6), 1378–1391.

Smith, S. L., Pavone, M., Bullo, F., & Frazzoli, E. (2010). Dynamic vehicle routing with priority classes of stochastic demands. *SIAM Journal on Control and Optimization*, *48*(5), 3224–3245.

Swihart, M. R., & Papastavrou, J. D. (1999). A stochastic and dynamic model for the single-vehicle pick-up and delivery problem. *European Journal of Operational Research*, *114*(3), 447–464.

Toth, P., & Vigo, D. (2002a). Models, relaxations and exact approaches for the capacitated vehicle routing problem. *Discrete Applied Mathematics*, *123*(1–3), 487–512.

Toth, P., & Vigo, D. (Eds.). (2002b). *The vehicle routing problem*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Waisanen, H. A., Shah, D., & Dahleh, M. A. (2008). A dynamic pickup and delivery problem in mobile networks under information constraints. *IEEE Transactions on Automatic Control*, *53*(6), 1419–1433.

Welch, P. D. (1983). The statistical analysis of simulation results. In S. S. Lavenberg (Ed.), *Computer performance modeling handbook*. New York, NY: Academic Press Inc. chap 6.

Zemel, E. (1984). Probabilistic analysis of geometric location problems. *Annals of Operations Research*, *1*(3), 215–238.

Pasquale Grippa received his Laurea (B.Sc.) degree and Laurea Specialistica (M.Sc.) degree with distinction in electronics engineering from the University of Salerno in 2007 and 2011, respectively. During his studies, he was guest student at TU Berlin and a research assistant at the Fraunhofer Heinrich Hertz Institute in Berlin. He is currently a staff member and Ph.D. student at University of Klagenfurt in the Institute of Networked and Embedded Systems. His interests include modeling, analysis, and control of multi-drone systems, decision making in autonomous transport, and phenomena of emergence and self-organization.

Doris A. Behrens Ph.D., has become Interim Modelling Unit Lead in early 2017 after being with ABCi for two years as Mathematical Modeller on a joint post with Aneurin Bevan University Health Board and Cardiff University. During this time, she has worked on models to forecast ED demand, issues of equity within the healthcare service to inform service redesign and the use of System Thinking to understand the Unscheduled Care system. Before Dr. Behrens developed a taste for improvement in healthcare she had led the Eastern European Centre of Competence in Austria and worked as Assistant Professor and Senior Researcher on a multitude of modelling/improvement projects, e.g. in Environmental Economics and Drug Control.

Friederike Wall earned her Diploma for Business Economics in 1988 and her Doctoral degree in 1991, both at the Georg-August-Univesität Göttingen, Germany. In 1996 she received the "venia legendi" (Habilitation for Business Economics) from the Universität Hamburg, Germany. After being a scientific Project Referent for Accounting in the context of the implementation of SAP R/3 at Max-Planck Society, Munich, Germany she became Full Professor of Business Administration, esp. Controlling and Information Management at the Universität Witten/Herdecke, Germany. Since 2009 she has been Full Professor and Head of the Department of Controlling and Strategic Management at the Alpen-Adria-Universität Klagenfurt, Austria where, since 2010, she also has been the vice rector of research. Prof. Wall's scientific work is focused on distributed decision-making, management control systems and the quality of the information provided by these systems. Her main research approach is defined by agent-based simulation methods and agent-based technologies.

Christian Bettstetter received the Dipl.-Ing. degree in 1998 and the Dr.-Ing. degree (summa cum laude) in 2004, both in electrical and information engineering from Technische Universität München (TUM), Munich, Germany. He was a research and teaching staff member at the Institute of Communication Networks, TUM, until 2003. From 2003 to 2005, he was a senior researcher with DOCOMO Euro-Labs. He has been a professor at the University of Klagenfurt, Austria, since 2005, and founding director of the Institute of Networked and Embedded Systems since 2007. He is also the founding scientific director of Lakeside Labs, a research company on self-organizing networked systems.