### ETHzürich



# **Energy-Efficient Fast Delivery by Mobile Agents**

Andreas Bärtschi

joint work with Thomas Tschager

ETH Department of Computer Science

# Model of delivery

### Setting

- undirected graph G = (V, E)with edges  $e \in E$  having lengths  $\ell_e$
- one package: source s and target t
- k agents, each starting at a node  $p_i$ , energy consumption  $w_i$  & velocity  $v_i$

## Assumptions

- agents cooperate by global, centralized coordination
- handovers possible at nodes V as well as inside edges



### Task

Find an efficient delivery in terms of energy consumption  $\mathcal{E} \longrightarrow$  terms of form  $w_i \cdot \ell_e$ delivery time  $\mathcal{T} \longrightarrow$  terms of form  $v_{i}^{-1} \cdot \ell_{P}$ 

FCT Bordeaux | September 13, 2017 | 2 / 11

Andreas Bärtschi

#### **ETH** zürich

### 1 Introduction

- Model of delivery
- Outline

### 2 Previous work

- Agents with weights only
- Agents with velocities only

### 3 Combining the two measures

- Combining energy- and time-efficiency
- OPT characterization
- Algorithm

## Agents with weights

Each agent has its individual energy consumption (weight)  $w_i$ . Can we optimize the total energy consumption  $\mathcal{E}$  needed to deliver the package?



not on shortest path

vertex handovers

### decreasing weights

[1] B., Chalopin, Das, Disser, Graf, Hackfeld, Penna: Energy-Efficient Delivery by Heterogeneous Mobile Agents

ETH Department of Computer Science Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 4 / 11

## Agents with velocities

Each agent has its individual velocity  $v_i$ .

Can we *optimize* the total *time*  $\mathcal{T}$  needed to deliver the package?



not on shortest path (multiple) in-edge-handovers

increasing velocities

[2] B., Graf, Mihalák: Collective fast delivery by energy-efficient agents

ETH Department of Computer Science Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 5 / 11

# Combining energy- and time-efficiency

Each agent has its individual weight  $w_i$  and velocity  $v_i$ . We want a delivery that is both efficient in its energy consumption and its delivery time:

- Fastest delivery among all energy-optimum ones.
   Task: lexicographically minimize (*E*, *T*).
- Energy-optimum delivery among all fastest ones.
   Task: lexicographically minimize (*T*, *E*).
- Tradeoff between the two measures. Task: minimize  $\varepsilon \cdot \mathcal{T} + (1 - \varepsilon) \cdot \mathcal{E}, \ \varepsilon \in (0, 1).$

 $\Rightarrow$  this talk

$$ightarrow$$
  $ightarrow$  NP-hard <sup>[2]</sup>

[2] B., Graf, Mihalák: Collective fast delivery by energy-efficient agents

ETH Department of Computer Science Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 6 / 11

# Agents with weights and velocities

Each agent has its individual *weight* w<sub>i</sub> and *velocity* v<sub>i</sub>. Among all energy-optimum deliveries, can we find the fastest?



# not on shortest path 0 or 1 in-edge-handovers decreasing tuples $(w_i, v_i^{-1})$

ETH Department of Computer Science

Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 7 / 11

# Characterization of an optimum delivery

## Theorem (OPT characterization)

There is an optimum delivery, with the involved agents denoted by  $1, \ldots, i, i + 1, \ldots, k$ , in which the following holds for each consecutive pair of agents:

• Decreasing weights:  $w_i \ge w_{i+1}$ .

• If 
$$w_i = w_{i+1}$$
, then  $v_i < v_{i+1}$ .

• If  $w_i = w_{i+1}$ , then agent i + 1 does not move without the package.

$$\underbrace{w_1 = w_2 = \dots}_{W_1} \xrightarrow{>} x \underbrace{v_i < v_{i+1} < \dots}_{W_i = w_{i+1} = \dots} \xrightarrow{>} y \underbrace{\dots = w_k}_{W_3}$$

 $\Rightarrow$  First look at the problem for each weight class  $W_i$  separately.

ETH Department of Computer Science Andreas

Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 8 / 11

#### **ETH** zürich

## Uniform energy consumption weights



Example: 4 agents, weight 3, velocities v₁ = 1, v₂ = 2, v₃ = 4, v₄ = 5. Approach:
Move closest agent to source s. Costs (E, T)[p₁] = (3 ⋅ 3, 3/1) = (9, 3).
Order agents by increasing velocity. Transform graph to DAG. Compute (E, T)[p<sub>i</sub>] = the energy consumption E[p<sub>i</sub>] and delivery time T[p<sub>i</sub>] of an optimum delivery of the package from s to p<sub>i</sub>, using only agents 1,..., i - 1.

ETH Department of Computer Science Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 9 / 11

#### **ETH** zürich

# Uniform energy consumption weights



Example: 4 agents, weight 3, velocities  $v_1 = 1$ ,  $v_2 = 2$ ,  $v_3 = 4$ ,  $v_4 = 5$ . Approach: 1 Move closest agent to source *s*. Costs  $(\mathcal{E}, \mathcal{T})[p_1] = (3 \cdot 3, 3/1) = (9, 3)$ . 2 Order agents by increasing velocity. Transform graph to DAG. Compute

$$\begin{aligned} & (\mathcal{E}, \mathcal{T})[p_2] = (\mathcal{E}, \mathcal{T})[p_1] + (5 \cdot 3, 5/1) = (24, 8). \\ & (\mathcal{E}, \mathcal{T})[p_3] = \min\{(\mathcal{E}, \mathcal{T})[p_1] + (9 \cdot 3, 9/1), \ (\mathcal{E}, \mathcal{T})[p_2] + (4 \cdot 3, 4/2)\} = (36, 10). \\ & (\mathcal{E}, \mathcal{T})[p_4] = \min\{\dots, (\mathcal{E}, \mathcal{T})[p_2] + (4 \cdot 3, 4/2), \dots\} = (36, 10). \end{aligned}$$

**3** Compute optimum *delivery time* among energy-optimum deliveries:  $(\mathcal{E}, \mathcal{T}) = (66, 12)$ . ETH Department of Computer Science Andreas Bärtschi FCT Bordeaux | September 13, 2017 | 9 / 11

## Vertex handovers



If  $w_i \neq 2w_{i+1}$ , then agent i does not handover the package to i+1 inside an edge. Assume we have  $w_i \neq 2w_i \ \forall i, j$ .

 $\Rightarrow$  We can use the precomputed weight class solutions. Define the subproblems

 $(\mathcal{E}, \mathcal{T})[j, y] =$  the energy consumption  $\mathcal{E}[j, y]$  and delivery time  $\mathcal{T}[j, y]$  of an optimum delivery of the package from s up to node y, using only agents from the first j weight classes  $W_1, \ldots, W_i$ .

 $\Rightarrow$  We can compute  $(\mathcal{E}, \mathcal{T})[j, y]$  from all smaller subproblems  $(\mathcal{E}, \mathcal{T})[j-1, x]!$ 

## In-edge handovers





We can have at most one in-edge-handover per edge.

But: Which agents are involved in an in-edge-handover?

Pareto frontier!

Adapt previous methods as follows:

Incorporate the Pareto frontier into the weight class computations.

Incorporate in-edge-handovers into the main dynamic program.

### **Running time:**

- Preprocessing  $\mathcal{O}(APSP + k + |V|)$ .
  Per weight class:  $\mathcal{O}(|V| \cdot k^2 + |V|^2 \cdot k)$ .
  Main dynamic program:  $\mathcal{O}(k \cdot |V| \cdot |V|)$ .

ETH Department of Computer Science

Andreas Bärtschi

FCT Bordeaux | September 13, 2017 | 11 / 11