7. Appendix

7.1. Proof of Theorem 11.

The proofs are a condensation and adaptation of the corresponding proofs in [34, 35, 54]. Changes are necessary since Algorithm 2 solves a different problem and uses different message passing updates and schedules than the algorithms from [34, 35, 54]. As a shorthand we will use $c_t^\lambda(y)$ instead of writing $\langle c_t^\lambda, y\rangle$ for a solution y of triangle subproblem $t \in T$.

Definition 12 (ϵ-optimal local solutions)**.** *For $e \in E$ define*

$$\mathcal{O}_e^\epsilon(\lambda) := \{x \in \{0,1\} : x \cdot c_e^\lambda \leq \min(0, c_e^\lambda) + \epsilon\} \quad (11)$$

and for $t \in T$

$$\mathcal{O}_t^\epsilon(\lambda) := \{x \in \mathcal{M}_T : c_t^\lambda(x) \leq \min_{x' \in \mathcal{M}_T} c_t^\lambda(x') + \epsilon\} \quad (12)$$

to be the ϵ- optimal local solutions.

Hence, $\mathcal{O}_e^0(\lambda) = \overline{c_e^\lambda}$ for $e \in E$ and likewise $\mathcal{O}_t^0(\lambda) = \overline{c_t^\lambda}$ for $t \in T$.

Definition 13 (ϵ-tolerance)**.** *The minimal value $\epsilon(\lambda)$ for which $\mathcal{O}^\epsilon(\lambda)$ has edge-triangle agreement is called called the ϵ-tolerance.*

Definition 14 (Algorithm Mappings)**.** *Let*

(i) $\mathcal{H}_{E\to T}(\lambda)$ be the Lagrange multipliers that result from executing lines 2-5 in Algorithm 2,

(ii) $\mathcal{H}_{T\to E}(\lambda)$ be the Lagrange multipliers that result from executing lines 8-13 in Algorithm 2,

(iii) $\mathcal{H} = \mathcal{H}_{T\to E} \circ \mathcal{H}_{E\to T}$ be one pass of Algorithm 2,

(iv) $\mathcal{H}^i(\cdot) = \underbrace{\mathcal{H}(\mathcal{H}(\ldots(\mathcal{H}(\cdot))\ldots))}_{i \text{ times}}$ be the i-fold composition of $\mathcal{H}$.

Note that $\mathcal{H}_{E\to T}$ and $\mathcal{H}_{T\to E}$ and consequently also $\mathcal{H}$ are well-defined mappings since, even though Algorithm 2 is parallel, the update steps do not depend on the order in which they are processed.

Lemma 15. *Let $\alpha \in (0,1]$ and let λ be Lagrange multipliers. Let $e \in E$ and $t \in T$ with $e \subsetneq t$. Define new Lagrange multipliers as*

$$\lambda'_{t',e'} = \begin{cases} \lambda_{t',e'} - \alpha c_e^\lambda, & e = e', t = t' \\ \lambda_{t',e'}, & e \neq e' \text{ or } t \neq t' \end{cases} \quad (13)$$

(i) $LB(c^\lambda) \leq LB(c^{\lambda'})$.

(ii) $\mathcal{O}_e(c^\lambda) \subseteq \mathcal{O}_e(c^{\lambda'})$.

(iii) $LB(c^\lambda) < LB(c^{\lambda'}) \Leftrightarrow \mathcal{O}_e(c^\lambda) \cap \Pi_{t,e}(\mathcal{O}_t(c^{\lambda'})) = \varnothing$.

(iv) $LB(c^\lambda) = LB(c^{\lambda'}) \Rightarrow \mathcal{O}_t(c^{\lambda'}) \subseteq \mathcal{O}_t(c^\lambda)$.

(v) $LB(c^\lambda) = LB(c^{\lambda'})$ and $c_e^\lambda \neq 0 \Rightarrow \Pi_{t,e}(\mathcal{O}_t(c^{\lambda'})) = \mathcal{O}_e(c^\lambda)$.

Proof. (i) If $c_e^\lambda \geq 0$ then $LB(c^\lambda)_e = LB(c^{\lambda'})_e$ and $LB(c^\lambda)_t \leq LB(c^{\lambda'})_t$ since $c_{t,e}^\lambda \leq c_{t,e}^{\lambda'}$.

If $c_e^\lambda < 0$ then $LB(c^\lambda)_e = c_e^\lambda < (1-\alpha)c_e^\lambda = LB(c^{\lambda'})_e$. $\leq LB(c^{\lambda'})_t = \min_{y\in\mathcal{M}_T} c^{\lambda'}(e) \geq \min_{y\in\mathcal{M}_T} c^\lambda(e) - \alpha c_e^\lambda = LB(c^\lambda)_t - \alpha c_e^\lambda$.

(ii) It holds that $c_e^{\lambda'} = (1-\alpha)c_e^\lambda$. Hence, if $\alpha = 1$ then $\mathcal{O}_e(c^\lambda) = \{0,1\}$ and the claim trivially holds. Otherwise $\mathcal{O}_e(c^\lambda) = \mathcal{O}_e(c^\lambda)$.

(iii) Assume $\mathcal{O}(c^\lambda)_t \cap \Pi_e(\mathcal{O}(c_t^{\lambda'}) = \varnothing$. Assume first that $\alpha = 1$. Then it must hold that $|\mathcal{O}(c^\lambda)_t| = 1$. Let $\{y_e^*\} = \mathcal{O}(c^\lambda)_e$ and $y_t^* \in \arg\min_{y\in\mathcal{M}_T} c_t^\lambda(y)$. Let $y'_e \in \arg\min_{y\in\{0,1\}} c_e^{\lambda'} y$ and $y'_t \in \arg\min_{y\in\mathcal{M}_T} c_t^{\lambda'}(y)$ such that $y'_e = \Pi_e(y'_t)$ (this is possible due to $\mathcal{O}(c^{\lambda'})_e = \{0,1\}$ for $\alpha = 1$. Then

$$\begin{aligned} LB(c^\lambda)_e + LB(c^\lambda)_t &= c_e^\lambda y_e^* + c_t^\lambda(y_e^*) \\ &< c_e^\lambda y'_e + c_t^\lambda(y'_e) \\ &= c_e^{\lambda'} y'_e + c_t^{\lambda'}(y'_e) = LB(c^{\lambda'})_e + LB(c^{\lambda'}) \,. \end{aligned} \quad (14)$$

For $\alpha < 1$ the result follows from the above and the concavity of LB.

Assume now $\mathcal{O}(c^\lambda)_t \cap \Pi_e(\mathcal{O}(c_t^{\lambda'}) \neq \varnothing$. Choose $y_e^* \in \mathcal{O}(c^\lambda)_e$ and $y_t^* \in \mathcal{O}(c^\lambda)_t$ such that $y_t^*(e) = y_e^*$. Then it holds that

$$\begin{aligned} LB(c^\lambda)_e + LB(c^\lambda)_t &= c_e^\lambda y_e^* + c_t^\lambda(y_t^*) \\ &= c_e^{\lambda'} y_e^* + c_t^{\lambda'}(y_t^*) > LB(c^{\lambda'})_e + LB(c^{\lambda'})_t \end{aligned} \quad (15)$$

Since LB is non-decreasing, it follows that $LB(c^\lambda) = LB(c^{\lambda'})$.

(iv) If $c_e^\lambda = 0$ there is nothing to show since $\lambda' = \lambda$.

Assume that $c_e^\lambda > 0$. Then it must hold that $0 \in \Pi_{t,e}(\mathcal{O}_t(c^\lambda))$ due to (iii). Since $c_t^{\lambda'}(e) > c_t^\lambda(e)$ and all other costs stay the same, it holds that

$$y_t \begin{cases} \in \mathcal{O}_t(c^{\lambda'}), & y_t \in \mathcal{O}_t(c^\lambda), y_t(e) = 0 \\ \notin \mathcal{O}_t(c^{\lambda'}), & y_t \notin \mathcal{O}_t(c^\lambda), y_t(e) = 0 \\ \notin \mathcal{O}_t(c^{\lambda'}), & y_t \in \mathcal{O}_t(c^\lambda), y_t(e) = 1 \\ \notin \mathcal{O}_t(c^{\lambda'}), & y_t \notin \mathcal{O}_t(c^\lambda), y_t(e) = 1 \end{cases} . \quad (16)$$

Hence, the result follows.

The case $c_e^\lambda < 0$ can be proved analoguously.

(v) Follows from the case by case analysis in (16) □

Lemma 16. *Let $\alpha \in (0,1]$ and let λ be Lagrange multipliers. Let $e \in E$ and $t \in T$ with $e \subsetneq t$. Define*

$$\lambda'_{t',e'} = \begin{cases} \lambda_{t',e'} + \alpha m_{t\to e}(c^\lambda_t), & e = e', t = t' \\ \lambda_{t',e'}, & e \neq e' \text{ or } t \neq t' \end{cases} \tag{17}$$

(i) $LB(c^\lambda) \leq LB(c^{\lambda'})$.

(ii) $\mathcal{O}_t(c^\lambda) \subseteq \mathcal{O}_t(c^{\lambda'})$.

(iii) $LB(c^\lambda) < LB(c^{\lambda'}) \Leftrightarrow \mathcal{O}_e(c^\lambda) \neq \Pi_{t,e}(\mathcal{O}_t(c^{\lambda'})$.

(iv) $LB(c^\lambda) = LB(c^{\lambda'}) \Rightarrow \mathcal{O}_e(c^{\lambda'}) \subseteq \mathcal{O}_e(c^\lambda)$

(v) $LB(c^\lambda) = LB(c^{\lambda'})$ *and* $m_{t\to e}(c^\lambda) \neq 0 \Rightarrow \Pi_{t,e}(\mathcal{O}_t(c^\lambda)) = \mathcal{O}_e(c^{\lambda'})$.

Proof. Analoguous to the proof of Lemma 15. □

Lemma 17. *Each iteration of Algorithm 2 is non-decreasing in the lower bound LB from (5).*

Proof. Follows from Lemma 15 (i) and Lemma 16 (i). □

Lemma 18. *If $LB(c^\lambda) = LB(\mathcal{H}(c^\lambda))$ then $\mathcal{O}_e(\mathcal{H}(c^\lambda)) \subseteq \mathcal{O}_e(c^\lambda)$ for all $e \in E$.*

Proof. If $\mathcal{O}_e(c^\lambda) = \{0,1\}$, there is nothing to show.

Assume $\{0\} = \mathcal{O}_e(c^\lambda)$. Then $\Pi_{t,e}(\mathcal{H}_{\mathrm{E}\to\mathrm{T}}(c^\lambda)_t) = \{0\}$ due to Lemma 15 (iv) for all $t \in T$, $e \subsetneq t$. Then Lemma 16 (v) implies that $\mathcal{O}_e(\mathcal{H}(c^\lambda)) = \{0\}$.

The case $\{1\} = \mathcal{O}_e(c^\lambda)$ can be proved analoguously. □

Lemma 19. *If $LB(c^\lambda) = LB(\mathcal{H}_{\mathrm{E}\to\mathrm{T}} \circ \mathcal{H}_{\mathrm{T}\to\mathrm{E}}(c^\lambda))$ then $\Pi_{t,e}(\mathcal{O}_t(\mathcal{H}_{\mathrm{E}\to\mathrm{T}} \circ \mathcal{H}_{\mathrm{T}\to\mathrm{E}}(c^\lambda))) \subseteq \Pi_{t,e}(\mathcal{O}_t(c^\lambda))$ for all $t \in T$, $e \in E$ and $e \subsetneq t$.*

Proof. Write $c^{\lambda'} = \mathcal{H}_{\mathrm{T}\to\mathrm{E}}(c^\lambda)$ and $c^{\lambda''} = \mathcal{H}_{\mathrm{E}\to\mathrm{T}}(c^{\lambda'})$. Let some $t \in T$, $e \in E$ and $e \subsetneq t$ be given. If $m_{t\to e} = 0$ the result follows from Lemma 15 (iv). Hence we can assume that $m_{t\to e} \neq 0$. Lemma 16 (iii) and (v) imply $\Pi_{t,e}(\mathcal{O}_t(c^\lambda)) = \mathcal{O}_e(c^{\lambda'})$. Due to Lemma 15 (iv) the result follows. □

Lemma 20. *Define $\xi_e = \mathcal{O}_e(c^\lambda)$ for all $e \in E$, $\xi_t = \mathcal{O}_t(\mathcal{H}_{\mathrm{E}\to\mathrm{T}}(c^\lambda))$, $\xi'_e = \mathcal{O}_e(\mathcal{H}(c^\lambda))$ for all $e \in E$ and $\xi'_t = \mathcal{O}_t(\mathcal{H}_{\mathrm{E}\to\mathrm{T}} \circ \mathcal{H}(c^\lambda))$. If $LB(c^\lambda) = LB(\mathcal{H}_{\mathrm{E}\to\mathrm{T}} \circ \mathcal{H}(c^\lambda))$ and ξ is not arc-consistent then $\exists e \in E$ such that $\xi'_e \subsetneq \xi_e$ or $\exists\, t \in T$, $e \in E$ and $e \subsetneq t$ such that $\Pi_{t,e}(\xi'_t) \subsetneq \Pi_{t,e}(\xi_t)$.*

Proof. If ξ is not arc-consistent there exists $e \in E$, $t \in T$ and $e \subsetneq t$ such that $\xi_e \neq \Pi_{t,e}(\xi_t)$.

The case $|\xi_e| = 1 = |\Pi_{t,e}(\xi_t)|$ implies $\xi_e \cap \Pi_{t,e}(\xi_t) = \varnothing$ and due to Lemma 16 (iii) contradicts that LB is not increasing.

Assume $\xi_e \subsetneq \Pi_{t,e}(\xi_t)$. Due to Lemma 15 (iv) and (v) this would imply an increase in the lower bound.

Hence we can assume that $\Pi_{t,e}(\xi_t) \subsetneq \xi_e$. Due to Lemma 16 (v) it holds that $|\xi'_e| = 1$.

Together with Lemmas 18 and 19 the result follows. □

Lemma 21. *$\mathcal{H}$ is a continuous mapping.*

Proof. All the operations used in Algorithm 2 are continuous, i.e. adding and subtracting, dividing by a constant and taking the minimum w.r.t. elements for the min-marginals. Hence, $\mathcal{H}$, the composition all such continuous operations, is continuous again. □

Lemma 22. *The lower bound LB from (5) is continuous in λ.*

Proof. Taking minima is continuous as well as addition. Hence LB is continuous as well. □

Lemma 23. *ϵ-tolerance is continuous in λ.*

Proof. We first prove that for any arc-consistent subset ξ the minimal ϵ for which $\xi \subseteq \mathcal{O}^\epsilon(\lambda)$ is continuous. To this end, note that the minimum ϵ such that $\xi_e \subseteq \mathcal{O}^\epsilon_e(\lambda)$ for any edge $e \in E$ can be computed as

$$\xi_e = \begin{cases} c^\lambda_e, & \xi = \{0\} \\ -c^\lambda_e, & \xi = \{1\} \\ |c^\lambda_e|, & \xi = \{0,0\} \end{cases} \tag{18}$$

All the expressions are continuous, hence the minimum ϵ for any edge is continuous. A similar observation holds for triangles. Since the ξ-specific ϵ is the maximum over all edges and triangles, it is continuous as well.

Since the ϵ-tolerance is the minimum over all minimal ξ-specific ϵ and there is a finite number of arc-consistent subsets ξ, the result follows. □

Lemma 24. *For any edge costs $c \in \mathbb{R}^E$ there exists $M > 0$ such that $\|\mathcal{H}^i(c)\| \leq M$ for any $i \in \mathbb{N}$.*

Proof. Assume $\mathcal{H}^i(c)$ is unbounded. If all $\mathcal{H}^i(c)_t$ $t \in T$ are bounded, all $\mathcal{H}^i(c)_e$ are bounded as well due to (6). Hence, there must exist $t \in T$ such that $\mathcal{H}^i(c)_t$ is unbounded. Since $LB(H^i)(c)_t$ is bounded below by Lemma 17 and trivially above by 0, it must hold that either

(i) there exists one edge $e \subsetneq t$ such that $\mathcal{H}^i(c)_t(e)$ converges towards $-\infty$ on a subsequence or

(ii) there exists at most one edge $e \subsetneq t$ such that $\mathcal{H}^i(c)_t(e)$ converges towards ∞ and there exist $e' \neq e'' \subsetneq t$ with $e' \neq e$ and $e'' \neq e$ such that $\mathcal{H}^i(c)_t(e')$ and $\mathcal{H}^i(c)_t(e'')$ converge towards $-\infty$ with $\mathcal{H}^i(c)_t(e) - \mathcal{H}^i(c)_t(e') \leq M'$ and $\mathcal{H}^i(c)_t(e) - \mathcal{H}^i(c)_t(e'') \leq M'$ where $M' > 0$ is a constant, since otherwise $LB(\mathcal{H}^i(c))_t$ would converge to $-\infty$.

Hence there must be at least double the number of Lagrange multipliers $\lambda_{t,e}$ that converge towards $-\infty$ than those that converge towards ∞ with at least the same rate. Hence, there must be $\tilde{e} \in E$ such that on a subsequence $\mathcal{H}^i(c)_{\tilde{e}}$ converges towards $-\infty$, contadicting that $LB(\mathcal{H}^i(c))_e$ is bounded below by Lemma 17. $\square$

Proof of Theorem 11. Due to the Bolzano Weierstrass theorem and the boundedness of $\mathcal{H}^i(c)$ there exists a subsequence $i(k)$ such that $\mathcal{H}^{i(k)}(c)$ converges to a c^{λ^*}. We first show that $\epsilon(c^{\lambda^*}) = 0$. Since $\mathcal{H}$ and LB are continuous an LB is non-decreasing, we have

$$\begin{aligned} LB(c^{\lambda^*}) &= \lim_{k\to\infty} LB(\mathcal{H}^{i(k)}(c)) \\ &= \lim_{k\to\infty} LB(\mathcal{H}^{i(k)+n}(c)) \quad \forall n \geq 0\,. \end{aligned} \tag{19}$$

Due to Lemma 20 and ϵ being continuous, $\epsilon(c^{\lambda^*}) = 0$ follows.

Define $s^i = \max_{j\leq i} \epsilon(\mathcal{H}^j(c))$. Then s^i is by construction a non-negative non-decreasing sequence and therefore has a limit s^*. Hence, there also must exist a subsequence $j(k)$ such that $\lim_{k\to\infty} \epsilon(\mathcal{H}^{j(k)}(c)) = s^*$. As proved above the subsequence $j(k)$ has a subsequence which converges towards $\epsilon(\cdot) = 0$, hence $s^* = 0$ as well. Finally,

$$0 \leq \epsilon(\mathcal{H}^i(c)) \leq s^i \tag{20}$$

implies convergence towards node-triangle agreement. $\square$

7.2. GPU implementations

Edge contraction We use a specialized implementation for edge contraction using Thrust [20] which is faster than performing it via general sparse matrix-matrix multiplication routines and most importantly has lesser memory footprint allowing to run larger instances. We store the adjacency matrix $A = (I, J, C)$ in COO format, where I, J, C correspond to row indices, column indices and edge costs resp. The pseudocode is given in Algorithm 4.

Algorithm 4: `GPU Edge-Contraction`

Data: Adjacency matrix $A = (I, J, C)$, Contraction mapping $f : V \to V'$
Result: Contracted adjacency matrix $A' = (I', J', C')$

```
// Assign new node IDs
Î(v) = I(f(v)), ∀v ∈ V
Ĵ(v) = J(f(v)), ∀v ∈ V
COO-Sorting(Î, Ĵ, C)
// Remove duplicates and add costs
(I', J', C') = reduce_by_key(
   keys=(Î, Ĵ),values=Ĉ, acc=+)
```

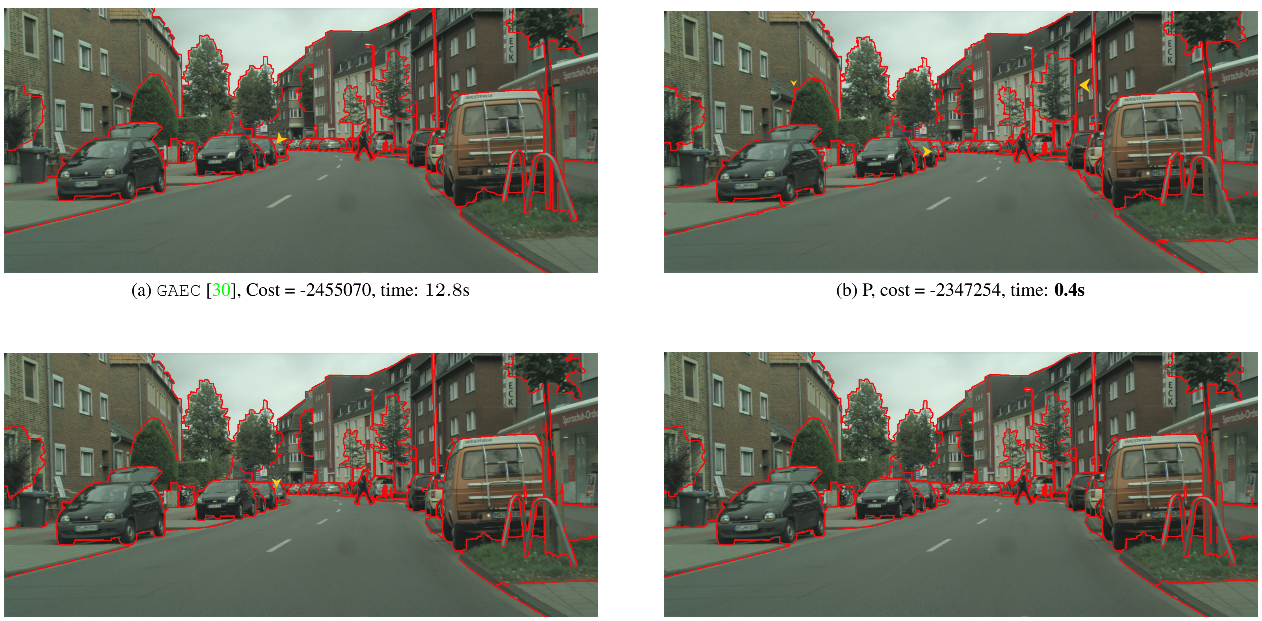
Conflicted cycles For detecting conflicted cycles we use specialized CUDA kernels. The pseudocode for detecting 5-cycles is given in Algorithm 5. The algorithm searches for conflicted cycles in parallel in the positive neighbourhood $\mathcal{N}^+$ of each negative edge. To efficiently check for intersection in Line 4 we store the adjacency matrix in CSR format.

Algorithm 5: `Parallel Conf. 5-Cycles`

Data: Adjacency matrix $A = (V, E, c)$
Result: Conflicted cycles Y in A

```
  // Partition edges based on costs
1 E⁺ = {ij ∈ E : c_ij > 0}
2 E⁻ = {ij ∈ E : c_ij < 0}
3 Y = ∅
  // Check for attractive paths
4 for v1v3 ∈ N⁺(v0) × N⁺(v4) : v0v4 ∈ E⁻ in
   parallel do
5   for v2 ∈ N⁺(v1) ∩ N⁺(v3) do
6     Y = Y ∪ {v0, v1, v2, v3, v4}
7   end
8 end
```

7.3. Results comparison



(a) GAEC [30], Cost = -2455070, time: 12.8s

(b) P, cost = -2347254, time: **0.4s**

(c) PD, cost = -2499152, time: 1.1s

(d) PD+, cost = $-$**2523547**, time: 2.2s

Figure 7. Results comparison on an instance of *Cityscapes* dataset highlighting the transitions. Yellow arrows indicate incorrect regions. Our purely primal algorithm (P) suffers in localizing the sidewalks and trees. PD+ is able to detect an occluded car on the left side of the road which all other methods did not detect. (Best viewed digitally)