TECHNISCHE UNIVERSITÄT WIEN
Institut für Computergraphik und Algorithmen

Two Approaches for Computing Lower Bounds on the Reconstruction of Strip Shredded Text Documents

Matthias Prandtstetter

Forschungsbericht / Technical Report

TR–186–1–09–01

August 2009

Favoritenstraße 9-11 / E186, A-1040 Wien, Austria
Tel. +43 (1) 58801-18601, Fax +43 (1) 58801-18699
www.cg.tuwien.ac.at

Two Approaches for Computing Lower Bounds on the Reconstruction of Strip Shredded Text Documents

Matthias Prandtstetter

Institute of Computer Graphics and Algorithms, Vienna University of Technology
Favoritenstraße 9–11/E1861, A-1040 Vienna, Austria

Abstract

In this paper we present two *integer linear programming* formulations for the reconstruction of strip shredded text documents. While the first formulation is based on well known cycle elimination constraints the second one is compact. Bounds are then computed by a *linear programming* relaxation as well as a *Lagrangian relaxation* approach. In addition a *Lagrangian heuristic* is introduced providing primal feasible solutions to the problem. Experimental results document the convincing performance of the proposed methods.

Key words: Document Reconstruction, Integer Linear Programming, Lagrangian Relaxation

1. Introduction

In the last years the reconstruction of strip shredded text documents (RSSTD) became more and more important—especially in the fields of investigative sciences and forensics. At the same time the machines used for destructing paper documents—so called shredders—were improved and the strips produced by them got thinner and thinner. Therefore, it is nearly impossible for humans to reconstruct strip shredded documents without computational assistance, which leads to a noticeably increased need for (semi-)automatic reconstruction systems.

Unfortunately, only a few approaches are published in literature concerning the automated reconstruction of shredded paper documents. De Smet et al. [1] propose methods for extracting features which are in the following used for matching two or more strips with each other. They mainly focus on methods derived from the large field of image processing and pattern recognition.

In contrast to this, Ukovich et al. [2] used a clustering algorithm on the given set of strips for identifying those strips forming one page. Unfortunately, no concrete sequence of strips is generated by this method.

In [3] we formulated RSSTD as a combinatorial optimization problem and proved that it is $\mathcal{NP}$-hard via a reduction to the generalized traveling salesman problem which is then further transformed into the symmetric traveling salesman problem for obtaining (heuristic) solutions to RSSTD. In this work, we also presented a hybrid method combining a *variable neighborhood search* (VNS) based approach with a system for integrating user actions into the search procedure.

Although all three of these methods differ in their basic concepts, none of these indicates how to evaluate the improvement potential in the quality of solutions provided by a computer, i.e., no lower bounds are given.

2. Problem Definition

Let us now introduce a formal problem definition of RSSTD: Given is a set $\mathcal{S} = \{1, \ldots, n-1\}$ of rectangular paper remnants $i \in \mathcal{S}$—so called strips—obtained as output of a shredding device. Each of the strips has the same height and written text or other valuable information printed on its front. We assume that the back of the strips is blank. As explained in [3] an additional blank strip n is added to $\mathcal{S}$ such that the cardinality of $\mathcal{S}$ is equal to n.

Any permutation of the strips in $\mathcal{S}$ such that the artificial strip n is placed at position n together with a binary vector indicating the orientations of the strips forms a feasible solution to RSSTD. We adopt the cost function $c_2(i, j, \omega_i, \omega_j)$ as presented in [3] for estimating the likelihood that strips $i, j \in \mathcal{S}$ did not appear side by side under orientations $\omega_i, \omega_j \in \mathcal{O}$, where $\mathcal{O}$ represents the set $\{\mathrm{d}, \mathrm{u}\}$ of possible orientations, with d denoting "down" and u denoting "up". An entire solution is evaluated by the total costs of the realized neighborhood relations. For short we write in the following $c(i, j, \omega)$, with $\omega \in \mathcal{O}^2$, i.e., $\omega \in \{(\mathrm{d}, \mathrm{d}), (\mathrm{d}, \mathrm{u}), (\mathrm{u}, \mathrm{d}), (\mathrm{u}, \mathrm{u})\}$, instead of $c_2(i, j, \omega_i, \omega_j)$. The goal of RSSTD is to reconstruct the original document pages, which were cut into the given set $\mathcal{S}$ of strips, i.e., to find a permutation of the strips and an associate orientation vector of minimum costs.

Email addresses: prandtstetter@ads.tuwien.ac.at (Matthias Prandtstetter)

3. Integer Linear Programming Model

In this section, we present two different *integer linear programming* (ILP) models for RSSTD. Let us assume that variable $s^{\omega}_{jj'} \in \{0,1\}$, with $1 \leq j, j' \leq n$ and $\omega \in \mathcal{O}^2$, is equal to 1 iff strip j' is the right neighbor of strip j and both are oriented according to ω. For the artificial strip n we define $s_{nj'} = 1$, iff strip j' is placed at position 1, i.e., the artificial strip is considered to be followed by the first strip. Using this variable definition the following model can be expressed, which provides a basis for the later proposed ILP formulations:

$$\min \sum_{j \in \mathcal{S}} \sum_{j' \in \mathcal{S}} \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{jj'} \cdot c(j, j', \omega) \tag{1.1}$$

$$\sum_{j' \in \mathcal{S}} \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{jj'} = 1, \qquad \forall\, j \in \mathcal{S} \tag{1.2}$$

$$\sum_{j \in \mathcal{S}} \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{jj'} = 1, \qquad \forall\, j' \in \mathcal{S} \tag{1.3}$$

$$\sum_{j' \in \mathcal{S}} s^{(\mathrm{d,u})}_{jj'} + s^{(\mathrm{d,d})}_{jj'} = \sum_{j' \in \mathcal{S}} s^{(\mathrm{u,d})}_{j'j} + s^{(\mathrm{d,d})}_{j'j}, \qquad \forall\, j \in \mathcal{S} \tag{1.4}$$

$$\sum_{j' \in \mathcal{S}} s^{(\mathrm{u,d})}_{jj'} + s^{(\mathrm{u,u})}_{jj'} = \sum_{j' \in \mathcal{S}} s^{(\mathrm{d,u})}_{j'j} + s^{(\mathrm{u,u})}_{j'j}, \qquad \forall\, j \in \mathcal{S} \tag{1.5}$$

$$\sum_{\omega \in \mathcal{O}} s^{\omega}_{jj'} + \sum_{\omega \in \mathcal{O}} s^{\omega}_{j'j} \leq 1, \qquad \forall\, j, j' \in \mathcal{S} \tag{1.6}$$

$$s^{\omega}_{jj} = 0, \qquad \forall\, j \in \mathcal{S},\ \omega \in \mathcal{O}^2 \tag{1.7}$$

$$s^{\omega}_{jj'} \in \{0,1\}, \qquad \forall\, \omega \in \mathcal{O}^2,\ j, j' \in \mathcal{S} \tag{1.8}$$

While the total costs for an assignment of strips to each other should be minimized according to expression (1.1), constraints (1.2) and (1.3) state that each strip j, with $1 \leq j \leq n$, has to be followed and preceded by exactly one strip, i.e., exactly one strip has to be assigned to the position right to strip j and one left to j. If a strip j precedes strip j' it is obvious that strip j follows another strip. Anyhow, the orientation of strip j has to be the same for both relations, see Eq. (1.4) and (1.5). As soon as one strip j is preceding another strip j' strip j cannot be placed right next to j', cf. Eq. (1.6).

Due to the strong relationship of RSSTD to (A)TSP it is obvious that optimal solutions with respect to formulation (1) can in general contain subtours, which are not valid for RSSTD. Therefore, we decided to implement and compare two different approaches for preventing subtours. The first one is based on cycle elimination constraints, which can be expressed as follows:

$$\sum_{k \in \mathcal{C}} \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{kk+1} \leq |\mathcal{C}| - 1, \qquad \forall\, \emptyset \neq \mathcal{C} \subset \mathcal{S}, \tag{2}$$

whereas $\mathcal{C}$ corresponds to cycles of length less than $|\mathcal{S}|$ and $k+1$ denotes the strip placed right to strip k on this cycles.

Since the number of constraints specified by expression (2) is exponential in the number of strips, an efficient separation of these constraints is necessary for computing practical results. This is done by first building a complete graph $G(V, E)$ whose nodes $v \in V$ correspond to strips. The weights of the edges $(i, j) \in E$ are computed as $\sum_{\omega \in \mathcal{O}^2} s^{\omega}_{ij}$, whereas the concrete values of variables s^{ω}_{ij} correspond to the current LP values. A new cut is found as soon as a circle is identified whose length is less than or equal to one.

The second approach for eliminating cycles is based on the introduction of additional variables $p_{ij} \in \{0,1\}$, with $1 \leq i, j \leq n$, whereas p_{ij} is equal to 1 iff strip j is assigned to position i and otherwise 0. Then the following constraints can be defined:

$$\sum_{i=1}^{n} p_{ij} = 1, \qquad \forall\, j \in \mathcal{S} \tag{3.1}$$

$$\sum_{j \in \mathcal{S}} p_{ij} = 1, \qquad \forall\, i = 1, \ldots, n \tag{3.2}$$

$$p_{1j'} = \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{nj'}, \qquad \forall\, j' \in \mathcal{S} \tag{3.3}$$

$$p_{n-1j} = \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{jn}, \qquad \forall\, j \in \mathcal{S} \tag{3.4}$$

$$p_{ij} + p_{i+1j'} - 1 \leq \sum_{\omega \in \mathcal{O}^2} s^{\omega}_{jj'}, \qquad \forall \begin{cases} i = 1, \ldots, n-2, \\ \quad j, j' \in \mathcal{S} \end{cases} \tag{3.5}$$

$$p_{nn} = 1 \tag{3.6}$$

$$p_{ij} \in \{0,1\}, \qquad \forall\, i = 1, \ldots, n,\ j \in \mathcal{S} \tag{3.7}$$

Due to constraints (3.1) and (3.2) it is assured that each strip is assigned to exactly one position and vice versa. Anyhow, a connection between variables p and s has to be established. This is done by Eq. (3.3), (3.4) and (3.5). If strip j is assigned to position i and strip j' to position $i+1$ then the according variables $s^{\omega}_{jj'}$, with $\omega \in \mathcal{O}^2$, have to be set to one. Finally, constraint (3.6) ensure that the artificial strip is assigned to position n.

In the further context, we will denote the two above presented formulations by *cycle elimination based formulation* (CEF) and *position assignment based formulation* (PAF), whereas CEF corresponds to the core formulation (1) amended by constraints (2) and PAF contain formulation (1) together with expressions (3). For practical results regarding the direct solution of these two formulations by using CPLEX we refer to Sec. 6.

4. Lagrangian Relaxation for RSSTD

Preliminary tests revealed that the application of exact approaches to RSSTD, e.g., a direct solution of CEF and PAF using general purpose ILP solvers, is limited to relatively small instances. Therefore, heuristic methods are of great importance when trying to solve real-world instances. Anyhow, one main drawback of many heuristics is the lack of providing (tight) bounds on the solution quality. To overcome this problem one could solve the *linear programming* (LP) relaxation of CEF or PAF; see Sec. 6 for computational results. In addition, we developed a *Lagrangian relaxation* (LR) [4] approach based on PAF. The main idea of LR is to substitute complicating constraints by corresponding penalty terms in the objective function. For this purpose, each relaxed constraint is associated with a so called *Lagrangian multiplier*. Subsequently, one tries to find a set of multipliers that maximizes the associated lower bound for the original minimization problem.

For this purpose we relax the linking constraints (3.3)–(3.5) of PAF resulting in the following new objective function:

$$\begin{aligned}\min \sum_{j\in\mathcal{S}}\sum_{j'\in\mathcal{S}}\sum_{\omega\in\mathcal{O}^2} & s_{jj'}^{\omega}\cdot c(j,j',\omega)+\\ \sum_{j'\in\mathcal{S}} & \lambda_{j'}^{1}\cdot\left(p_{1j'}-\sum_{\omega\in\mathcal{O}^2} s_{nj'}^{\omega}\right)+\\ \sum_{j\in\mathcal{S}} & \lambda_{j}^{2}\cdot\left(p_{n-1j}-\sum_{\omega\in\mathcal{O}^2} s_{jn}^{\omega}\right)+\\ \sum_{i=1}^{n-2}\sum_{j\in\mathcal{S}}\sum_{j'\in\mathcal{S}} & \lambda_{i,j,j'}^{3}\cdot\left(p_{ij}+p_{i+1j'}-1-\sum_{\omega\in\mathcal{O}^2} s_{jj'}^{\omega}\right)\end{aligned} \tag{4}$$

After applying some basic transformations and substituting (constant) expressions by (newly introduced) coefficients ρ_{ij}, $\sigma_{jj'}^{\omega}$ and δ, with $1\leq i,j,j'\leq n$ and $\omega\in\mathcal{O}^2$, the LR approach can be formulated as follows:

$$\min \underbrace{\sum_{j\in\mathcal{S}}\sum_{i=1}^{n-1}(\pi_{ij}\cdot p_{ij})}_{\text{SP I}}+\underbrace{\sum_{j\in\mathcal{S}}\sum_{j'\in\mathcal{S}}\sum_{\omega\in\mathcal{O}^2}\left(\sigma_{jj'}^{\omega}\cdot s_{jj'}^{\omega}\right)}_{\text{SP II}}+\delta \tag{5}$$

subject to Eq. (1.2)–(1.8), (3.1), (3.2), (3.6), and (3.7)

with

$$\rho_{1j}=\lambda_j^1+\sum_{j'=1}^{n}\lambda_{i,j,j'}^{3}+\sum_{j'=1}^{n}\lambda_{i,j',j}^{3}, \qquad \forall\ j\in\mathcal{S} \tag{6}$$

$$\rho_{ij}=\lambda_j^2+\sum_{j'=1}^{n}\lambda_{i,j,j'}^{3}+\sum_{j'=1}^{n}\lambda_{i,j',j}^{3}, \qquad \forall\begin{cases} i=2,\dots,n-2\\ j\in\mathcal{S}\end{cases} \tag{7}$$

$$\rho_{n-1j}=\lambda_j^2+\sum_{j'=1}^{n}\lambda_{n-1,j',j}^{3}, \qquad \forall\ j\in\mathcal{S} \tag{8}$$

$$\sigma_{j,j'}^{\omega}=c(j,j',\omega)-\sum_{i=1}^{n-2}\lambda_{i,j,n}^{3}, \qquad \forall\ \begin{cases} \omega\in\mathcal{O}^2\\ j,j'\in\mathcal{S}\setminus\{n\}\end{cases} \tag{9}$$

$$\sigma_{n,j}^{\omega}=c(n,j,\omega)-\lambda_j^1-\sum_{i=1}^{n-2}\lambda_{i,n,j}^{3}, \qquad \forall\ \begin{cases} \omega\in\mathcal{O}^2\\ j\in\mathcal{S}\setminus\{n\}\end{cases} \tag{10}$$

$$\sigma_{j,n}^{\omega}=c(j,n,\omega)-\lambda_j^2-\sum_{i=1}^{n-2}\lambda_{i,j,n}^{3}, \qquad \forall\ \begin{cases} \omega\in\mathcal{O}^2\\ j\in\mathcal{S}\setminus\{n\}\end{cases} \tag{11}$$

$$\sigma_{n,n}^{\omega}=c(n,n,\omega)-\lambda_n^1-\lambda_n^2-\sum_{i=1}^{n-2}\lambda_{i,n,n}^{3}, \qquad \forall\ \omega\in\mathcal{O}^2 \tag{12}$$

$$\delta=-\sum_{j\in\mathcal{S}}\sum_{j'\in\mathcal{S}}\sum_{i=1}^{n-2}\lambda_{i,j,j'}^{3} \tag{13}$$

Based on the fact, that the coefficients π, σ and δ are composed of linear combinations of λ^1, λ^2, λ^3 and the cost function c, see Eq. (6)–(13), it can be observed that the above formulation decomposes into two independent subproblems only linked by the objective function (5). The first subproblem SP I formulated via variables p_{ij}, with $1\leq i\leq n-1$ and $j\in\mathcal{S}$, corresponds to a linear assignment problem. It is well known that this problem can be efficiently solved. The second subproblem SP II formulated via variables $s_{jj'}^{\omega}$, with $j,j'\in\mathcal{S}$ and $\omega\in\mathcal{O}^2$, corresponds to a *generalized traveling salesman problem* [5] allowing subtours of arbitrary lengths. Although the non-generalized version of this subproblem, i.e., the classical *traveling salesman problem* allowing subtours, could be easily solved using bipartite matching algorithms, it can be easily shown that the integrality property does not hold for SP II, which implies that bounds provided by our LR approach might be better than those provided by an LP relaxation of PAF [4].

For computing lower bounds by means of LR, we implemented a standard subgradient method as described in [4] by initializing all Lagrangian multipliers to 0 and setting the strategic parameter $\pi=2$. The value of π is halved as soon as 30 subgradient iterations without improvement on the lower bound were performed. In contrast, π is doubled when an improvement could be achieved and $\pi\leq 1$ holds. This iterative process is terminated once π falls below 0.001 or the lower bound provided by this method corresponds to the best known upper bound, which is iteratively updated based on the solutions generated by the Lagrangian heuristic presented within the next section. For solving subproblems SP I and SP II we directly applied the general purpose ILP solver CPLEX 11.2. We refer to Sec. 6 for detailed results including a comparison of bounds obtained via LP relaxations and those obtained by using LR.

5. A Lagrangian Heuristic

Based on the LR presented in the previous section, we developed a *Lagrangian heuristic* (LH) which provides feasible solutions to the original problem based on the values of the relaxed ILP. The main idea is to decode the neighborhood relations and orientations of strips such that a feasible solution is generated. Since the absolute positions of strips, i.e., the values of variables p_{ij}, are not necessarily consistent with the relative positions, i.e., the values of variables $s_{jj'}^{\omega}$, we decided to neglect the information about the absolute position within this decoding step and derive a feasible solution from the relative positions only, which also primarily contribute to the objective function. Since the virtual strip n is placed at the last position (see Eq. (3.6)), we start the decoding by placing this strip at position n. According to the values of s_{jn}^{ω}, with $1\leq j\leq n-1$ and $\omega\in\mathcal{O}^2$, we place that strip $\overline{j}$ at position $n-1$ which has a corresponding variable $s_{\overline{j}n}^{\omega}$ equal to 1. Of course, the orientation of the strip is also regarded. This method is applied iteratively as long as not already positioned strips are concerned. In the case of a cycle, we restart the method by placing a randomly chosen and so far not positioned strip at the last yet free position.

Since any permutation of strips with the artificial strip placed at the last position forms a valid solution, this method always provides feasible solutions. Further, by using appropriate datastructures the runtime of this approach is in $O(n^2)$ as for each position at most $4n$ variables have to be evaluated.

6. Experimental Results

To evaluate the performances and the contributions of the above presented approaches, we applied them to real-world instances of RSSTD. For generating instances, we used those documents introduced by Ukovich et al. in [2], which were then converted into grayscale images and were (virtually) cut into 80 to 135 strips, each. These settings correspond to strip widths of 2.6mm to 1.5mm. The test results presented within this section were obtained on a single core of an Intel QuadCore 5150 with 2.8GHz and 8GB RAM and ILOG CPLEX 11.2 has been used as general purpose (I)LP solver.

For computing lower bounds by means of LR we implemented the standard subgradient method, whereas the upper bound is updated based on the solutions provided by the proposed LH. The Lagrangian multipliers were all initialized to 0. Obviously, the execution of the subgradient method is aborted as soon as the lower and upper bound are identical. We analyzed the bounds provided by LR and the LP relaxation of CEF on 560 instances in total and the main result is that in most cases, i.e., in 517 out of 560, the obtained bounds are equal. Only for 43 instances of which all where generated based on the first document page of the test set introduced by Ukovich et al. a difference in the quality of the bounds could be identified. The corresponding results are shown in Tab. 1, whereas the first column indicates the number of strips the page was cut into and the second column lists the absolute objective values of the original document pages. The remaining numbers should be interpreted as follows: the columns labeled with LR present the (relative) bounds obtained using our LR approach in relation to the objective of the original document page as well as the number of iterations until the LR was terminated. In case the number of iterations is equal to one the solution derived by our LH approach by setting all Lagrangian multipliers to zero, i.e., solving the core formulation (1) solely, is proven optimal. The last column of Tab. 1 lists the (relative) bounds provided by directly solving the LP relaxation of CEF.

The following two observations can be made based on the test results: first of all the bounds obtained by our LR approach are equal or better than the bounds provided by an LP formulation using subcycle elimination constraints. We assume, however, that this behavior is mainly based on the objective function used for estimating the likelihood of placing two strips next to each other. Furthermore we expect to emphasize this positive property of our cost function when considering more problem specific information by calculating the concrete cost values, e.g., by considering the character orientations, applying optical character recognition (OCR), or incorporating the likelihood that two patterns identified on the corresponding strip edges match with each other. In that case we assume that the error made by the cost function is even further minimized.

Table 1: Results comparing the bounds obtained by the proposed LR and the LP relaxation of CEF in relation to the original document page (orig.). In addition the number of LR iterations until LR was terminated are provided.

		LR		
strips	orig.	bound	iter.	CEF
80	29408	99.8232 %	(1)	99.5103 %
81	29408	99.8232 %	(1)	99.5103 %
86	31494	99.6444 %	(1)	99.4253 %
87	31494	99.6444 %	(1)	99.4253 %
88	31494	99.6444 %	(1)	99.4253 %
89	32774	99.8047 %	(1)	99.6400 %
90	32774	99.8047 %	(1)	99.6400 %
91	32440	100.0000 %	(1)	99.8243 %
92	32440	100.0000 %	(1)	99.8243 %
93	32440	100.0000 %	(1)	99.8243 %
96	36256	100.0000 %	(1)	99.7849 %
97	36256	100.0000 %	(1)	99.7849 %
98	36256	100.0000 %	(1)	99.7849 %
106	37122	99.9407 %	(1)	99.9003 %
107	37122	99.9407 %	(1)	99.9003 %
108	37122	99.9407 %	(1)	99.9003 %
109	38694	99.8346 %	(331)	99.8217 %
110	38694	99.8346 %	(331)	99.8217 %
111	38694	99.8346 %	(331)	99.8217 %
112	38694	99.8346 %	(331)	99.8217 %
113	39836	99.9699 %	(1)	99.8519 %
114	39836	99.9699 %	(1)	99.8519 %
115	39836	99.9699 %	(1)	99.8519 %
116	39836	99.9699 %	(1)	99.8519 %
117	39926	99.8397 %	(331)	99.8197 %
118	39926	99.8397 %	(331)	99.8197 %
119	39926	99.8397 %	(331)	99.8197 %
120	39962	99.8398 %	(331)	99.8123 %
121	42422	99.7737 %	(1)	99.6818 %
122	42422	99.7737 %	(1)	99.6818 %
123	42422	99.7737 %	(1)	99.6818 %
124	42422	99.7737 %	(1)	99.6818 %
125	42454	99.7739 %	(1)	99.6773 %
126	44682	99.8836 %	(1)	99.8590 %
127	44682	99.8836 %	(1)	99.8590 %
128	44682	99.8836 %	(1)	99.8590 %
129	44728	99.8837 %	(2)	99.8748 %
130	44728	99.8837 %	(2)	99.8748 %
131	45698	99.9912 %	(1)	99.8534 %
132	45698	99.9912 %	(1)	99.8534 %
133	45698	99.9912 %	(1)	99.8534 %
134	45698	99.9912 %	(1)	99.8534 %
135	45698	99.9912 %	(1)	99.8534 %

The second conclusion which can be drawn from the results is that the number of iterations until our LR approach terminates is typically low. In most cases there is even only one iteration. For some instances, however, it was not possible to improve the bound obtained during the first iteration of LR, but at the same time LH was not able to provide a primal feasible solution with identical objective value. Again, we expect to improve on this issue

Table 2: Comparison of computation times and solution qualities of PAF and CEF when directly solved using CPLEX 11.2. Numbers without parentheses indicate running times in seconds until the optimal solution was obtained (including optimality proof) whereas numbers in parentheses indicate the relative gap of current best integer and best dual bounds after 1200 seconds of computation time.

strips	20	30	40	50	60	70	80	90	100
inst.	PAF/CEF	PAF/CEF	PAF/ CEF	PAF/ CEF	PAF/ CEF	PAF/ CEF	PAF/ CEF	PAF/CEF	PAF/ CEF
p01	0.3/ 0.3	2.0/ 0.6	(0.04)/(0.07)	(0.50)/(0.02)	(0.01)/(0.10)	(0.80)/(0.01)	(0.80)/(0.01)	1200.2/ 42.5	(0.81)/ 68.2
p02	0.5/ 0.1	7.9/ 0.3	311.1/ 0.7	(0.52)/ 1.2	(0.83)/ 7.8	(0.84)/ 4.0	(0.84)/ 6.8	(0.83)/ 19.8	(0.83)/ 149.1
p03	0.2/ 0.1	72.2/ 0.6	1.3/ 0.3	30.7/ 0.6	(0.44)/ 1.1	(0.77)/ 2.4	102.6/ 2.7	(0.80)/ 4.1	(0.80)/ 4.2
p04	0.1/ 0.1	7.3/ 0.2	18.7/ 0.2	117.5/ 0.6	(0.77)/ 1.3	(0.12)/ 0.9	(0.16)/ 2.8	(0.11)/ 28.1	(0.80)/ 5.6
p05	0.1/ 0.1	0.5/ 0.1	1.2/ 0.5	101.8/ 0.2	(0.18)/ 2.4	249.9/ 1.2	290.4/ 0.7	(0.56)/ 7.1	(0.71)/(0.06)
p06	1.1/ 0.0	67.0/ 0.2	1.1/ 0.4	(0.18)/ 0.7	(0.29)/ 0.7	(0.09)/ 1.9	148.9/ 5.2	813.1/ 23.2	720.5/ 4.7
p07	0.1/ 0.1	0.4/ 0.1	4.0/ 0.5	281.9/ 0.2	(0.34)/ 1.8	(0.20)/ 1.1	(0.41)/ 0.8	(0.69)/ 2.4	(0.77)/ 17.3
p08	0.2/ 0.1	0.8/ 0.1	108.4/ 0.8	7.7/ 0.2	(0.42)/ 0.8	(0.20)/ 1.0	(0.75)/ 0.8	(0.76)/ 4.4	174.1/ 2.5
p09	0.2/ 0.1	0.8/ 0.3	147.3/ 0.7	(0.24)/ 1.0	(0.21)/ 0.9	676.7/ 0.5	(0.64)/ 1.6	150.8/ 2.5	(0.76)/ 2.5
p10	0.5/ 0.1	1.8/ 0.3	306.6/ 0.3	(0.78)/ 1.1	637.1/ 2.0	(0.78)/ 2.6	(0.79)/ 4.7	(0.78)/ 14.4	(0.78)/ 6.2

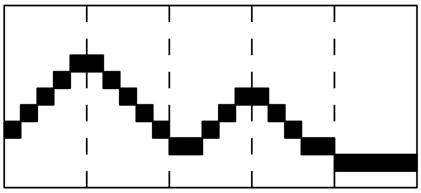

Figure 1: If this "document" is torn along the dashed lines, not all Lagrangian multipliers are set to zero in the set of optimal multipliers when using the LR approach.

by adapting the cost function as already indicated above.

Based on this observation the initialization of the Lagrangian multipliers to zero seems not only to be valuable but to be the only reasonable approach for providing good bounds as well as solving RSSTD. Nevertheless, not for all instances all Lagrangian multipliers are set to zero in the optimal set of multipliers. See for example the document shown in Fig. 1. When tearing this page along the dashed lines some multipliers have to be set to values not equal to zero for eliminating the cycles implied by the first two strips as well as the third and the fourth strip.

In addition to the experiments listed in Tab. 1 we tested to directly solve the ILP formulations presented in Sec. 3 via CPLEX. The corresponding results are listed in Tab. 2. For this test setting we used again the document pages introduced by Ukovich et al. This time, however, they were cut into 20 to at most 100 strips each, since preliminary tests revealed that the direct application of the general purpose ILP solver CPLEX to the above presented ILP formulations can be very time-consuming and for more than 110 strips the computation times did in most cases exceed a given time limit of 1200 seconds.

The numbers presented in Tab. 2 should be interpreted as follows. We present for each document page (p01–p10) and number of strips (20–80) the time (in seconds) until the optimal solution was found (and its optimality was proven). In case the optimal solution was either not reached or was not proven to be optimal within 1200 seconds of available computation time we present the relative gap of the so far best found integer solution and the dual bound computed by CPLEX in parentheses.

As can be seen, the numbers in Tab. 2 show that by directly applying CPLEX to the two ILP formulations, CEF leads to far better results than PAF. In concrete, for almost all instances with 50 or more strips optimal solutions could be obtained via CEF in some seconds of computation time. For only a few instances of that sizes even CEF could not lead to proven optimal solutions. Furthermore, for those instances with less than 50 strips, CEF provided more often the optimal solution and even in case both formulations could achieve optimality the computation times for the approach based on CEF where in most cases shorter.

Although Tab. 2 implies that solving a model based on CEF via CPLEX is much more efficient, the bounds obtained via the LR/LH approach are a little bit more promising than the results computed by the LP relaxation of CEF. Since the runtimes until the bounds were achieved did relatively strongly vary for both approaches no clear statement can be given which of the two different approaches for computing dual bounds is in the given case faster. Nevertheless, both the LR/LH approach and the computations of LP relaxations provide a good toolkit for producing valuable (lower) bounds. Furthermore, the LH often provides the optimal solution within a few iterations of the LR/LH approach.

7. Conclusions and Future Work

In this paper we presented two different ILP formulations strongly related to formulations for the traveling salesman problem. Whereas the first formulations called *position assignment based formulation* (PAF) has a polynomially bounded number of constraints and variables the second formulation called *cycle elimination based formulation* (CEF) relies on well known cycle elimination constraints. For providing lower bounds, we introduced a *Lagrangian relaxation* (LR) approach based on PAF including a *Lagrangian heuristic* (LH) providing primal feasible solutions based on the values computed by our LR approach.

Experimental results showed that the approaches for computing lower bounds were really successful and often provided the best obtainable lower bound. Further, the LH frequently generates optimal solutions based on the lower bounds presented by the LR method. We also showed that the bounds provided by LR are in some cases better than those obtained via a *linear programming* (LP) relaxation of CEF, although in most cases both approaches achieved the same bound. We could, however, not decide which approach for computing lower bounds is in general faster since the runtimes strongly vary for both approaches.

With respect to the solution of RSSTD CEF seems to be much better when directly solved using CPLEX 11.2. The size of the instances, i.e., the number of strips, which can be solved using this exact approach is, however, limited to relatively small instances.

Our tests for computing lower bounds were limited on instances with up to 135 strips which is due to the fact that in general the computation times grow fast as well as the memory consumption exceeds the limits of currently available standard hardware when trying to provide bounds on larger instances. Therefore it is necessary to further investigate subproblem SP II of our LR approach such that solutions to this problem could be computed even for large instances of RSSTD.

In addition, the numbers presented in Tab. 1 reveal that in most cases the optimal solution with respect to objective function (1.1) does not correspond with the original document page, i.e., the LR approach terminated by obtaining a primal solution via LH with an objective value identical to the bound computed by LR. We assume, however, that by using a more sophisticated function for estimating the likelihood of a positive match of two strips it is possible to overcome this problem, e.g., line spacings on two (possibly) matched strips could be incorporated into the objective function.

Finally, it has to be emphasized that any objective function will suffer from the fact that the final decision whether or not two strips match has to be done by humans and for each likelihood estimation a counter example forming the worst case, i.e., a false positive or false negative match, can be created. Therefore, as already shown in [3], it is essential that any automatic method for reconstructing strip shredded text documents is combined with a user interface integrating human interactions. Of course, this needs also to be incorporated when computing bounds on RSSTD why we will further examine different ways for sharing information between automatic systems for reconstructing documents and human operators.

References

[1] P. D. Smet, J. D. Bock, W. Philips, Semiautomatic reconstruction of strip-shredded documents, in: A. Said, J. G. Apostolopoulos (Eds.), Image and Video Communications and Processing 2005, Vol. 5685(1) of Proceedings of SPIE, SPIE, San Jose, CA, USA, 2005, pp. 239–248.

[2] A. Ukovich, A. Zacchigna, G. Ramponi, G. Schoier, Using clustering for document reconstruction, in: E. R. Dougherty, *et al* (Eds.), Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning, Vol. 6064 of Proceedings of SPIE, International Society for Optical Engineering, 2006, pp. 168–179.

[3] M. Prandtstetter, G. R. Raidl, Combining forces to reconstruct strip shredded text documents, in: M. J. Blesa, et al. (Eds.), Hybrid Metaheuristics 2008, Vol. 5269 of LNCS, Springer-Verlag Berlin Heidelberg, 2008, pp. 175–189.

[4] J. E. Beasley, Lagrangian Relaxation, in: C. R. Reeves (Ed.), Modern heuristic techniques for combinatorial problems, John Wiley & Sons, Inc., New York, NY, USA, 1993, pp. 243–303.

[5] C. Feremans, M. Labbe, G. Laporte, Generalized network design problems, European Journal of Operational Research 148 (1) (2003) 1–13.