Higher Order Matching for Consistent Multiple Target Tracking

Chetan Arora   Amir Globerson
School of Computer Science and Engineering
The Hebrew University
http://www.cs.huji.ac.il/~chetan/

Abstract

This paper addresses the data assignment problem in multi frame multi object tracking in video sequences. Traditional methods employing maximum weight bipartite matching offer limited temporal modeling. It has recently been shown [6, 8, 24] that incorporating higher order temporal constraints improves the assignment solution. Finding maximum weight matching with higher order constraints is however NP-hard and the solutions proposed until now have either been greedy [8] or rely on greedy rounding of the solution obtained from spectral techniques [15]. We propose a novel algorithm to find the approximate solution to data assignment problem with higher order temporal constraints using the method of dual decomposition and the MPLP message passing algorithm [21]. We compare the proposed algorithm with an implementation of [8] and [15] and show that proposed technique provides better solution with a bound on approximation factor for each inferred solution.

1. Introduction

Popularity of tracking by detection approaches [2] has led to a renewed interest in the data assignment problem in computer vision. In tracking by detection, a detection algorithm is first applied independently to find objects of interest in all frames. In the second step various detections across frames are associated with each other. This is typically done by associating a score with each such assignment, and finding the assignment with a maximum score. A good score function should capture the plausibility of an assignment. For example, low scores may be given to matching pairs which are visually dissimilar or are detected far from each other. In a crowded scenario such scores fail to disambiguate the correct assignment from other possible assignments. In these cases, one requires more complex scores. For example, scores which consider the velocity vectors implied by a matching, and constrain those to be physically valid are recommended for such cases. These

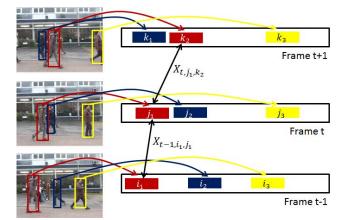are the scores we focus on in the current paper.

Figure 1: Higher Order Matching (campus sequence [2]).

Whatever the score function, one needs an algorithm for finding the optimal assignment. If the scores factor as a sum over individual assignments, then the problem can be solved via network flow algorithms [4, 25] or as a sum of bipartite matchings [23] defined over set of every two consecutive frames. However, such scores are not sufficiently descriptive, as they do not enforce more global properties of valid assignments, such as roughly constant velocity. To model such velocity constraints, one needs to consider pairs of assignments (i.e., a score which depends on three frames simultaneously), and the maximization problem becomes NP hard. We refer to such assignment problems with constraints involving more than 2 frames as *higher order assignment/matching* problems.

Several approximate maximization algorithms have recently been proposed to address NP hardness of higher order assignment problems [6–8, 15]. Leordeanu and Hebert [15] relax the integrality and matching constraints (a detection in one frame must be assigned to exactly one detection each in previous and next frame). They show that an optimal solution to the relaxed problem corresponds to the eigenvector with largest eigenvalue of suitably created symmetric matrix. Since the solution obtained may not be

feasible, they employ a greedy rounding scheme which iteratively removes the conflicting variables to generate a feasible assignment solution.

Collins [8] proposed a block ICM based technique for assignment problems with constraints involving two or more frames. Unlike [15], his method maintains a feasible solution at every step, and converges to a local minimum. It is similar to the iterated conditional modes (ICM) algorithm, but is applied at each step to a block of variables representing possible associations between two consecutive frames. The block-optimal conditional mode at each step is calculated as the solution to a bipartite matching problem.

Butt and Collins [6] have proposed to solve a series of independent higher order matching problems over frame triplets which are then merged into longer trajectories. The approach does not have the ability to revisit and correct a trajectory. Their method is designed specifically for frame triplets. Additionally there is no bound on approximation factor of the solution available with any of the discussed approaches [6, 8, 15].

The problem of matching in a arbitrarily long sequence with constraints involving 3 frames can also be formulated as 3-matching problem defined over a T-partite graph (T is the overall number of frames). A nice introduction on equivalent formulations can be found in [8]. 3-matching has been a popular problem in algorithmic community where the problem is shown to be APX-complete, that is, it is hard to approximate it within some constant factor [3].

Duchi et al. [10] have suggested a method called COMPOSE for optimizing matching problems with additional scores on pairs of edges. Their approach works by iteratively solving matching and mincut problems. It is in fact an implementation of the max-product algorithm for this particular setting. However, applying it to our problem with T frames would require minimizing cuts over graphs of size $O(T)$ which can be costly for large T. The approach we propose here scales only linearly in T for each update.

Given the above, our goal was to develop better approximation algorithms for the higher order assignment problem. It turns out that the dual decomposition (DD) framework (see below) is a perfect fit for this problem, and provides several desirable properties. First, it is scalable and each iteration is linear in the number of frames. Second, it offers both upper and lower bounds on the optimal scores, which can be used to obtain optimality certificates in some cases. Third, it can be naturally extended to other higher order scores involving three or more frames. Finally, it outperforms all previous methods considered for this task.

The DD approach (see [5] Sec. 6.4 for a general overview and [14, 21] for applications to inference) is conceptually simple: it takes a complex score function and breaks it down into a sum of scores that can be efficiently optimized. These problems are then modified using *messages* such that the

sum of the separate maximizations yields an upper bound on the true max. Finally, the messages are optimized such that the bound is as tight as possible. Many algorithms for optimizing the messages exist, and typically involve simple local updates. Here we use the MPLP approach [21] which works out nicely for our setup.

Poore [19] and Deb et al. [9] have suggested a Lagrangian relaxation scheme that is related to dual decomposition [20]. However, their objective is more involved and the message passing scheme we suggest is considerably simpler than their algorithm. Recently, Butt and Collins [7] applied Lagrangian relaxation to an objective similar to the one we use here. However, the resulting algorithm is different from ours, since it needs to solve a complete flow problem in each iteration, and uses subgradient updates which typically converge more slowly than coordinate descent [e.g., see 16].

The paper is structured as follows: we first present the higher order assignment problem in Section 2 followed by a brief review of the DD approach in Section 3. Next, in Section 4 we show how to apply the DD approach to our problem, and describe the resulting MPLP algorithm in Section 5, with its exactness results in Section 6. Finally, in Section 7 we provide experiments that demonstrate the utility of our approach. Specifically, we show that the proposed algorithm outperforms state of the art approaches [8, 15], yielding higher scoring assignments on various publicly available datasets [1, 2, 11], while also providing upper and lower bounds on the optimal score.

2. Problem Setup

We begin by formulating the score maximization problem. Denote the frames by $1, \ldots, T$. To simplify presentation, we assume that at each frame we have D detections indexed by $1, \ldots, D$. To these, we add a *dummy* detection at index 0 which handles partial trajectories. The goal is to find a set of paths from detections in the first frame to those in the last frame. Each such path corresponds to a single moving object.

Following [8], we note that a set of trajectories may be encoded via the union of all edges in the paths. We represent these paths via a set of boolean variables $X_{t,i,j} \in \{0, 1\}$ (with $t \in \{1, \ldots, T-1\}$ and $i, j \in \{0, \ldots, D\}$) where $X_{t,i,j} = 1$ iff there is an edge between detection i in frame t and detection j in frame $t+1$. See Figure 1.

Since the X variables correspond to a set of disjoint paths, they must satisfy the constraint that each detection in frame t is assigned to a single detection in frame $t+1$, and vice versa. This constraint need not hold for the dummy detection 0, which is meant to absorb partial paths. Thus the

X should satisfy:

$$\sum_{i=0}^{D} X_{t,i,j} = 1 \quad \forall t \in \{1, \dots, T-1\}, j \in \{1, \dots, D\}$$

$$\sum_{j=0}^{D} X_{t,i,j} = 1 \quad \forall t \in \{1, \dots, T-1\}, i \in \{1, \dots, D\}.$$

The set of X that satisfy the above constraints will be denoted by $\mathcal{M}$.

Next, we wish to construct a score function that maps each X to a number indicating how likely the proposed assignment is.[1]

The first element in the cost function considers each variable $X_{t,i,j}$ separately. Assume we have a weight $\bar{W}_{t,i,j}$ that is high if $X_{t,i,j}$ is a likely edge. The corresponding contribution to the score function is $\bar{W}_{t,i,j} X_{t,i,j}$.

Such local score functions are useful, but do not represent more global properties of the assignment. For example, since we know that objects tend to move in straight lines, it makes sense to give higher scores to X assignments that correspond to such trajectories, as suggested in [8]. To evaluate a change in movement direction, three frames are needed. Thus, we add the element $\bar{W}_{t,i,j,k} X_{t,i,j} X_{t+1,j,k}$ where $\bar{W}_{t,i,j,k}$ is high if detections i, j, k in frames $t, t+1, t+2$ approximately lie on a line.

The overall score function is then:

$$S(X) = \sum_{t,i,j} \bar{W}_{t,i,j} X_{t,i,j} + \sum_{t,i,j,k} \bar{W}_{t,i,j,k} X_{t,i,j} X_{t+1,j,k}. \tag{1}$$

This can be simplified, by absorbing the local scores into the pairwise ones. Define:

$$W_{t,i,j,k} = \bar{W}_{t,i,j,k} + \frac{1}{2} \left[\bar{W}_{t,i,j} + \bar{W}_{t+1,j,k} \right]. \tag{2}$$

Since every pairwise score includes exactly one edge from previous and next layers the two formulations are equivalent. The score can then be rewritten as:

$$S(X) = \sum_{t,i,j,k} W_{t,i,j,k} X_{t,i,j} X_{t+1,j,k}. \tag{3}$$

The overall optimization problem is:

$$\max_{X \in \mathcal{M}} S(X). \tag{4}$$

As mentioned earlier, the problem is equivalent to what is better known in the theory community as the 3-matching problem. Problems of the above form are known to be NP hard. In fact, the 3-matching problem belongs to the Karp's list of 21 NP-complete problems [13]. When $\bar{W}_{t,i,j,k} = 0$

---

[1]We do not construct a probabilistic model here, but it is possible to do so, as in [25].

and only local costs are considered, then the problem becomes easy since it can be separated into T separate bipartite matching constraints. However, introducing the higher order scores makes the problem considerably more complicated, requiring approximate solution approaches. In what follows we describe a simple and effective scheme for pairwise scores, which can be generalized to other higher order score functions as well.

3. Dual Decomposition

Dual decomposition (DD) is a powerful method for approximating discrete optimization problems. We present a brief review of DD that largely follows [21]. Consider a set of discrete variables $X = X_1, \dots, X_n$. Assume we have a set of functions $\theta_f(X)$. The functions typically depend on only a subset of the variables X (e.g., $\theta_f(X)$ may depend only on X_2, X_3). We denote the scope of each θ_f by S_f (e.g., $S_f = \{2, 3\}$ in the previous example). For notational convenience we write $\theta_f(X)$ instead of $\theta_f(X_{S_f})$.

We are interested in maximizing the sum of all these functions, namely we wish to maximize $\theta(X)$ defined as:

$$\theta(X) = \sum_f \theta_f(X). \tag{5}$$

We denote the above maximum value by θ^*.

DD is meant to address cases where maximizing the above sum is a hard problem (e.g., NP hard) *but* maximizing each $\theta_f(X)$ (or similarly structured functions) individually is easy. The idea is to construct a bound on the max value and tighten this bound. Specifically, we define a set of dual variables $\delta_{fi}(X_i)$ for each factor f, each variable $i \in S_f$ and each value X_i (e.g., in the example above we have $\delta_{f2}(X_2), \delta_{f3}(X_3)$). These dual variables may be thought of as a message from factor f to variable i, indicating a *prior* on the value X_i.

For a given δ we define a new set of factor functions (often known as reparameterizations):

$$\theta_f^{\delta}(X) = \theta_f(X) - \sum_i \delta_{fi}(X_i), \tag{6}$$

and a new set of *singleton* factor functions:[2]

$$\theta_i^{\delta}(X_i) = \sum_f \delta_{fi}(X_i). \tag{7}$$

Next, define the following *dual* function $L(\delta)$:

$$L(\delta) = \sum_i \max_{X_i} \theta_i^{\delta}(X) + \sum_f \max_X \theta_f^{\delta}(X). \tag{8}$$

It is easy to see that $L(\delta)$ upper bounds the θ^* value for all values of δ. It is thus sensible to minimize $L(\delta)$ w.r.t. δ, which is precisely what the DD framework proposes.

---

[2]Note that in the original $\theta(X)$ we did not have singleton factors.

The function $L(\delta)$ may be minimized using a variety of approaches. One that is particularly simple and effective is to use block coordinate descent on the δ variables. There are many schemes for doing this. Here we use the MPLP algorithm [21] which fixes all messages except those from a particular f to all variables i. The non-fixed messages are then updated to the value minimizing $L(\delta)$, which can be done in closed form. The updates are given by:

$$\delta_{fi}(X_i) = -\delta_i^{-f}(X_i) + \frac{1}{|f|} \max_{X \backslash X_i} \left[\theta_f(X_f) + \sum_{i \in f} \delta_i^{-f}(X_i) \right],$$
$$(9)$$

where $|f|$ denotes the number of variables in the factor θ_f, and we used $\delta_i^{-f}(X_i)$ to denote the sum of messages into i that are not from f. Namely:

$$\delta_i^{-f}(X_i) = \sum_{\bar{f} \neq f} \delta_{\bar{f}i}(x_i). \qquad (10)$$

The update in Eq. (9) is performed simultaneously for all messages from f to its variables. This is guaranteed to monotonically decrease the objective $L(\delta)$ [21]. Since $L(\delta)$ is not strictly-convex this scheme is not guaranteed to reach a global optimum [see 5, for discussion of convergence for coordinate descent]. However, this is often not an issue, and can be rectified via smoothing if needed [16] (smoothing is also helpful for accelerating sub gradient descent approaches, as proposed in [18] and applied to DD in [12, 17]).[3]

Eventually, we are interested in an assignment for X. This is typically done by taking the arg max of $\theta_i^\delta(X_i)$. Any such decoded assignment X provides a natural lower bound on θ^*, namely $\theta(X)$. Thus, if the upper bound $L(\delta)$ and the lower bound coincide, we know we have found the θ^* value and maximizing assignment.

4. Dual Decomposition for Higher Order (HO) Matching

We begin by rewriting Eq. (4) as a sum of relatively simple functions. Our functions will combine the matching constraints with the score elements from $S(X)$.

For convenience, we define a function $s_{t,i}(X)$ that contains the pairwise scores [4] corresponding to the ith detection in the tth frame:

$$s_{t,i}(X) = \sum_{j,k} W_{t-1,j,i,k} X_{t-1,j,i} X_{t,i,k}, \qquad (11)$$

so that:

$$S(X) = \sum_{t,i} s_{t,i}(X). \qquad (12)$$

---

[3]The smoothing approach is easily applicable in our case. We do not pursue it here since the non-smoothed version already performs well.

[4]Pairwise score in the formulation refers to the score corresponding to matching a triplet in three adjacent frames.

Next, define a function $\theta_{t,i}(X)$ that has a value of $-\infty$ if the ith detection in the tth frame violates the matching constraint.[5] Otherwise $\theta_{t,i}(X)$ has the value corresponding to the score $S(X)$ for this detection. Namely:

$$\theta_{t,i}(X) = \begin{cases} s_{t,i}(X) & \sum_j X_{t-1,j,i} = 1, \sum_j X_{t,i,j} = 1 \\ -\infty & \text{Otherwise.} \end{cases}$$
$$(13)$$

Finally, define:

$$\theta(X) = \sum_{t,i} \theta_{t,i}(X) \qquad (14)$$

Then it's easy to see that Eq. (4) is equivalent to:

$$\max_X \theta(X). \qquad (15)$$

We have thus turned Eq. (4) into a maximization of a sum of functions, as in the DD objective of Eq. (5), where f in Eq. (5) corresponds to a pair of indices (t, i) in Eq. (14). We next show how DD and the MPLP algorithm can be applied to this decomposition.
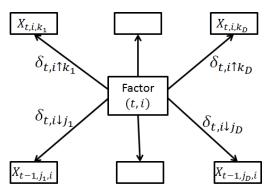
5. MPLP for Higher Order (HO) Matching

Figure 2: Factor for MPLP based HO Matching

To write the DD objective for Eq. (14), we introduce dual variables for messages between each factor (t, i) and the variables that participate in this factor. Recall that the factor (t, i) depends on the variables $X_{t,i,j}$ (i.e, matchings between frame t and $t+1$) and $X_{t-1,j,i}$ (i.e., matchings between frame $t-1$ and frame t). To reduce notational clutter we denote the message between factor (t, i) and $X_{t,i,j}$ by $\delta_{t,i\uparrow j}(X_{t,i,j})$ and the message between factor (t, i) and $X_{t-1,j,i}$ by $\delta_{t,i\downarrow j}(X_{t-1,j,i})$ (see figure 2).

Now define the reparameterized functions (see Eq. (6) and Eq. (7)):

$$\theta_{t,i}^\delta(X) = \theta_{t,i}(X) - \sum_j \delta_{t,i\uparrow j}(X_{t,i,j}) - \sum_j \delta_{t,i\downarrow j}(X_{t-1,j,i}),$$
$$(16)$$

---

[5]The ith detection in t^{th} frame must be matched with exactly 1 detection in $(t-1)^{th}$ and $(t+1)^{th}$ frames.

and:

$$\theta_{t,i,j}^{\delta}(X_{t,i,j}) = \delta_{t,i\uparrow j}(X_{t,i,j}) + \delta_{t+1,j\downarrow i}(X_{t,i,j}). \quad (17)$$

The dual $L(\delta)$ is therefore:

$$\theta(X) = \sum_{t,i} \max_X \theta_{t,i}^{\delta}(X) + \sum_{t,i,j} \max_{X_{t,i,j}} \theta_{t,i,j}^{\delta}(X_{t,i,j}). \quad (18)$$

We now turn to the MPLP updates in Eq. (9). The max operation in these updates involves all variables in $\theta_{t,i}$, namely $2D$ variables (assuming D matching pairs in each two consecutive frames). The cost of maximizing over all their assignments thus seems exponential at first. However, we note that $\theta_{t,i}$ is non-infinite only for $O(D^2)$ assignments satisfying the matching constraints, making the MPLP updates tractable. We first define $W'_{t,k,i,j}$ to be the value inside the brackets of Eq. (9) for the case $X_{t-1,k,i} = X_{t,i,j} = 1$, and all other variables of type $X_{t-1,\cdot,i}$ and $X_{t,i,\cdot}$ as zero.[6] This turns out to be:

$$\begin{aligned} W'_{t,k,i,j} = & \ W_{t-1,k,i,j} - \delta_{t+1,j\downarrow i}(1) - \delta_{t-1,k\uparrow i}(1) \\ & - \sum_{k'\neq k} \delta_{t-1,k'\uparrow i}(0) - \sum_{j'\neq j} \delta_{t+1,j'\downarrow i}(0) \quad (19) \end{aligned}$$

Next, we note that the argmax in the MPLP update must correspond to such a case (namely that exactly two variables are 1). Thus we conclude:

$$\begin{aligned} \delta_{t,i\uparrow j}(1) &= -\delta_{t+1,j\downarrow i}(1) + \frac{1}{2D} \max_k W'_{t,k,i,j} \\ \delta_{t,i\uparrow j}(0) &= -\delta_{t+1,j\downarrow i}(0) + \frac{1}{2D} \max_{k,j'\neq j} W'_{t,k,i,j'} \quad (20) \end{aligned}$$

Similarly:

$$\begin{aligned} \delta_{t,i\downarrow j}(1) &= -\delta_{t-1,j\uparrow i}(1) + \frac{1}{2D} \max_k W'_{t,j,i,k} \\ \delta_{t,i\downarrow j}(0) &= -\delta_{t-1,j\uparrow i}(0) + \frac{1}{2D} \max_{k,j'\neq j} W'_{t,j',i,k} \quad (21) \end{aligned}$$

The above MPLP updates monotonically decrease $L(\delta)$, providing an upper bound on the MAP. To obtain an assignment from δ we consider the singleton scores $\theta_{t,i,j}^{\delta}(X_{t,i,j})$ and return a matching that maximizes these. Namely, we solve:[7]

$$\arg\max_{X\in\mathcal{M}} \sum_{t,i,j} \theta_{t,i,j}^{\delta}(X_{t,i,j}) \quad (22)$$

This can be solved efficiently by solving a maximum weight bipartite matching independently for each consecutive frames t and $t+1$. The overall algorithm is provided in Algorithm 1.

---

[6]This corresponds to a matching between k and i in frames $t-1, t$ respectively, and between i and j in the frames $t, t+1$ respectively.

[7]Note that the simple MPLP decoding scheme will not add the constraint $X \in \mathcal{M}$. However, maximizing explicitly over $\mathcal{M}$ as in Eq. (22) makes sense, since the optimal X is constrained to be in $\mathcal{M}$.

6. Exactness for Local Scores

As with any approximation scheme, it is interesting to ask when our method will provide an exact answer. In what follows, we show that when the scores are only local, our method is exact. In other words, we consider the case that $\bar{W}_{t,i,j,k} = 0$ (see Section 2 for notation). As mentioned earlier, in this case, the maximization of $S(X)$ simply turns into T separate bipartite matching problems and can therefore be solved efficiently. However, it is not immediately clear that our DD scheme returns an exact solution in this setting. We show this below.

Recall that in the above case we have that $W_{t,i,j,k}$ is given by (ignoring the 0.5 factor):

$$W_{t,i,j,k} = \bar{W}_{t,i,j} + \bar{W}_{t+1,j,k} \quad (23)$$

We next simplify the DD objective in Eq. (18) for this parameter setting. The maximization $\max_X \theta_{t,i}^{\delta}(X)$ here is particularly simple since it breaks down into two separate maximizations (for the previous and next frames). So $\max_X \theta_{t,i}^{\delta}(X)$ turns out to be:

$$\begin{aligned} & \max_j \left[\bar{W}_{t-1,j,i} - \delta_{t,i\downarrow j}(1) + \delta_{t,i\downarrow j}(0) \right] - \sum_j \delta_{t,i\downarrow j}(0) + \\ & \max_j \left[\bar{W}_{t,i,j} - \delta_{t,i\uparrow j}(1) + \delta_{t,i\uparrow j}(0) \right] - \sum_j \delta_{t,i\uparrow j}(0) \end{aligned}$$

Given this simplified form, we can now take the dual of the minimization in Eq. (18). To obtain a dual, we first turn the minimization into a constrained problem by adding variables $\xi_{t,i\uparrow}, \xi_{t,i\downarrow}$ and constraints:

$$\begin{aligned} \xi_{t,i\downarrow} &\geq \bar{W}_{t-1,j,i} - \delta_{t,i\downarrow j}(1) + \delta_{t,i\downarrow j}(0) \quad \forall j \\ \xi_{t,i\uparrow} &\geq \bar{W}_{t,i,j} - \delta_{t,i\uparrow j}(1) + \delta_{t,i\uparrow j}(0) \quad \forall j \\ \xi_{t,i,j} &\geq \delta_{t,i\uparrow j}(X_{t,i,j}) + \delta_{t+1,j\downarrow i}(X_{t,i,j}) \quad \forall X_{t,i,j} \end{aligned}$$

The DD objective is then to minimize:

$$\sum_{t,i} \left[\xi_{t,i\downarrow} + \xi_{t,i\uparrow} \right] - \sum_{t,i,j} \delta_{t,i\uparrow j}(0) - \sum_{t,i,j} \delta_{t,i\downarrow j}(0) + \sum_{t,i,j} \xi_{t,i,j} \quad (24)$$

subject to the constraints above. We now take the dual of this LP. Introduce dual variables $\mu_{t,i\downarrow j}, \mu_{t,i\uparrow j}, \mu_{t,i,j}$ for the three sets of constraints above.[8] In deriving the dual we actually obtain that $\mu_{t,i\uparrow j} = \mu_{t+1,j\downarrow i} = \mu_{t,i,j}$, namely only the $\mu_{t,i,j}$ variables are needed. The dual then simplifies to (up to factor 2):

$$\begin{aligned} \max \quad & \sum_{t,i,j} \bar{W}_{t,i,j} \mu_{t,i,j} \\ \text{s.t.} \quad & \sum_j \mu_{t,i,j} = 1, \ \sum_j \mu_{t,j,i} = 1, \mu \geq 0 \end{aligned} \quad (25)$$

---

[8]For the third constraint there are actually variables $\mu_{t,i,j}(0)$ and $\mu_{t,i,j}(1)$, but the first can be eliminated, so we denote the latter simply by $\mu_{t,i,j}$.

Algorithm 1 HO Matching Algorithm

Input: Weights $W_{t,i,j,k}$ specifying the score for matching detections i, j, k in frames $t-1, t, t+1$.
Output: A set of binary variables $X_{t,i,j}$ specifying a valid matching.
Initialize: For all $t, i, j, X_{t,i,j}$ initialize $\delta_{t,i\uparrow j}(X_{t,i,j}) = 0$ and $\delta_{t+1,j\downarrow i}(X_{t,i,j}) = 0$.

 1: **while** Change in dual is not small enough **do**
 2:     **for** All factors t, i in a random order **do**
 3:         Calculate $W'_{t,k,i,j}$ as in Eq. (19) for all k, j.
 4:         Update messages $\delta_{t,i\uparrow j}(0), \delta_{t,i\uparrow j}(1)$ and $\delta_{t,i\downarrow j}(0), \delta_{t,i\downarrow j}(1)$ for all j as in Eq. (20) and Eq. (21).
 5:     **end for**
 6: **end while**
 7: Return the matching X that solves Eq. (22).

First, note that the above LP can be solved separately for each t (since there is no interaction between different t). Second, the LP for each t is in fact precisely the LP formulation of bipartite matchings, which is known to have an integral solution, and return the maximum bipartite matching (e.g., see Section 2.3 in [22]). Thus we conclude the minimum of the DD objective has the value of the optimal matching. Furthermore, it can be shown (see [21] section 1.7.2) that if MPLP converges to this value and the optimal matching is unique, then our decoding procedure Eq. (22) will find this optimal matching.

Finally, we emphasize that our procedure will in practice return the exact matchings in many other cases, where higher order factors are not zero. In the following empirical results we indeed observe several such cases.

7. Experiments

We next compare the proposed algorithm with self developed implementations of block ICM [8] and Spectral [15]. Since Spectral requires eigenvalue decomposition and scales quadratically with T as opposed to our method and block ICM, we evaluate it only on the short toy problem sequences. In contrast block ICM as well as our proposed approach scales well over long sequences which is the subject of our second experiment. It may be noted that there could have been algorithms other than MPLP based approach used for solving the proposed problem formulation in this paper. We have also tried subgradient descent, and found that it was substantially slower than our approach, and depended heavily on initial step size. Accelerated subgradient descent (for the smoothed objective) [12, 18] is likely to outperform standard subgradient. However, empirical results in [16] show that coordinate descent outperforms accelerated gradient (although this is of course problem dependent). The comparison with subgradient based approach is therefore not included in our experiments.

Our first evaluation is on simple problems, constituting the first 3 frames of the publicly available datasets TUD [2], ETH [11] and PSU [1]. Table 1 shows the compara-

tive results. The local scores have been calculated based upon the Euclidean distance between the detections. The pairwise scores are set as the distance between the detection in the middle frame and the centroid of detections in the first and third frames of the frame triplet (constant velocity assumption). This is one instance of higher order scores, and other scores utilizing appearance based cues could have been used. However, the purpose of experiments is to study the inference capabilities of the various algorithms with higher order matching constraints when the appearance based cues are ambiguous. Indicatory scores consonant with the objective have been used accordingly without compromising the generality of the algorithmic approach. Accordingly the quality of solution is measured in terms of primal value obtained which better indicates the inference capability of the compared algorithms instead of more standard mismatch error percentage which may be affected by tuning chosen/given weights.

The result of experiments on the toy problem is shown in Table 1. Both block ICM and the proposed MPLP perform much better than the spectral technique. Furthermore, in $5/9$ cases, MPLP finds a provably optimal solution (since the upper and lower bounds coincide).

Our second set of experiments focused on large problem sets containing complete sequences from various sources. Table 2 lists the results. Due to our experience in the toy problem and the scalability issues with Spectral approach we compare only to the block ICM approach. The local and pairwise scores have been set similarly as in toy problem case. Except for a short sequence (ETH Seq 2) MPLP outperforms block ICM approach on all tested datasets. One possible explanation for the results could be that the technique in [8] iterates through hard assignments as opposed to the message passing style of our method. It is thus more likely to get stuck in local minima than ours when the sequence is longer or the triplet weight is higher which is consistent with the observations in Table 1 and 2. Additionally MPLP is able to certify optimality in certain PSU sequences.
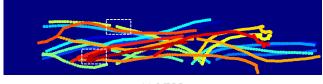
Figure 3 gives a visual comparison between MPLP and

Sequence	MPLP Primal	MPLP Dual	Block ICM Primal	Spectral Primal
TUD Campus	22505.59	23268.42	22505.59	17226.44
TUD Crossing	20325.02	20325.02	20325.02	6839.90
ETH Seq 1	7036.49	9558.66	6413.70	9513.79
ETH Seq 2	16031.69	23901.19	17011.80	10833.55
ETH Seq 3	25913.67	25913.67	25913.67	13605.31
ETH Seq 4	17730.87	17730.87	17730.87	14035.53
PSU Seq 1	13373.29	13373.29	13373.29	10051.84
PSU Seq 2	47747.79	47747.79	47747.79	35253.92
PSU Seq 3	36481.29	46298.73	25535.66	19679.38

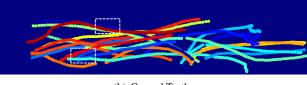Table 1: Comparison on first 3 frames of various sequences

Sequence	Num frames	MPLP Primal	MPLP Dual	Block ICM Primal
TUD Campus	71	915728.84	921786.11	907632.37
TUD Crossing	201	3213609.80	3390158.46	3203062.07
ETH Seq 01	56	490044.79	637833.04	472306.89
ETH Seq 02	9	130782.85	197381.71	148222.87
ETH Seq 03	999	33281471.13	34741881.37	33228228.12
ETH Seq 04	446	6884950.87	7024285.86	6880832.90
PSU Seq 01	61	708549.75	708549.75	708549.75
PSU Seq 02	121	5863545.86	5865261.80	5859410.28
PSU Seq 03	21	621002.00	678703.03	596113.92

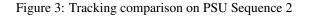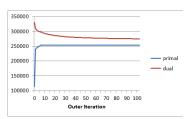Table 2: MPLP and Block ICM comparison on complete sequences

(a) MPLP

(b) Ground Truth

(c) Block ICM

Figure 3: Tracking comparison on PSU Sequence 2

block ICM on a PSU sequence. All trajectories are color-coded with the matched detections appearing in same color. The assignment differences between MPLP and block ICM have been marked with white rectangles. MPLP performs better than block ICM in the presence of strong matching ambiguities arising due to multiple close detections.

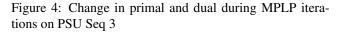Figure 4: Change in primal and dual during MPLP iterations on PSU Seq 3

Figure 4 shows an instance of the upper and lower bounds reported by MPLP. We show primal and dual values after different outer iterations (an outer iteration processes each factor exactly once) of MPLP on a test dataset (PSU Seq 3). As the iterations proceed, the quality/score of primal solutions keep increasing while the upper bound on the optimal primal given by the value of dual keeps tight-

ening. The approximation factor improves in both the cases with the number of iterations. In this problem instance, the bounds do not meet and we cannot conclude that the solution is optimal. However, in many cases (e.g., Table 1) the bounds do coincide and we obtain a certificate of optimality.

8. Conclusion

We presented an approach for optimizing higher order assignment problems that arise in the context of tracking by detection. Our approach relies on the dual decomposition framework which breaks the difficult assignment problem into simpler tractable tasks. We showed the inference capability of the algorithm in the presence of pairwise matching scores arising from detections in 3 consecutive frames. Such scores can successfully capture the constant velocity assumption which is a useful assignment cue in crowded scene when local scores are ambiguous. The strength of the algorithm is its simplicity. The algorithm is efficient and can scale well to long sequences. Our empirical results indicate that the proposed algorithm outperforms the state of the art compared with two recently introduced baselines.[9]

The DD message passing framework is very general, and thus we expect it will be effective for other higher order factors that are introduced into the tracking problem. For example, one may consider the problems with constraints involving 4 or more frames. Such constraints can encode, for example, acceleration (improving over constant velocity assumption in this paper) or statistical similarity between detections.

Finally, the structure of the algorithm is natural for providing results in an online manner. As new frames arrive, we can perform a small number of message passes for the most recent frames, to obtain upper and lower bounds for the overall sequence.

References

[1] PSU HUB Dataset. Last visited: Feb 24, 2013. `http://www.cse.psu.edu/~rcollins/software.html`.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008.

[3] G. Ausiello, M. Protasi, A. Marchetti-Spaccamela, G. Gambosi, P. Crescenzi, and V. Kann. *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer-Verlag, 1st edition, 1999.

[4] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *TPAMI*, 33(9), 2011.

[5] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, 1995.

[6] A. A. Butt and R. T. Collins. Multiple target tracking using frame triplets. In *ACCV*, 2012.

[7] A. A. Butt and R. T. Collins. Multi-target tracking by Lagrangian relaxation to min-cost network flow. In *CVPR*, pages 1846–1853, 2013.

[8] R. Collins. Multitarget data association with higher-order motion models. In *CVPR*, 2012.

[9] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom. A generalized s-d assignment algorithm for multisensor-multitarget state estimation. *IEEE Trans. on Aerospace and Electronic Systems*, 33(2):523–538, 1997.

[10] J. Duchi, D. Tarlow, G. Elidan, and D. Koller. Using combinatorial optimization within max-product belief propagation. In *NIPS*. 2007.

[11] A. Ess, B. Leibe, K. Schindler, , and L. van Gool. A mobile vision system for robust multi-person tracking. In *CVPR*, 2008.

[12] V. Jojic, S. Gould, and D. Koller. Accelerated dual decomposition for MAP inference. In *ICML*, pages 503–510, 2010.

[13] R. M. Karp. Reducibility Among Combinatorial Problems. In *Complexity of Computer Computations*, pages 85–103. 1972.

[14] N. Komodakis, N. Paragios, and G. Tziritas. MRF energy minimization and beyond via dual decomposition. *TPAMI*, 33(3):531–552, 2011.

[15] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *ICCV*, 2005.

[16] O. Meshi, T. Jaakkola, and A. Globerson. Convergence rate analysis of MAP coordinate minimization algorithms. In *NIPS*. MIT Press, 2012.

[17] I. Necoara and J. A. K. Suykens. Application of a smoothing technique to decomposition in convex optimization. *IEEE Trans. on Automatic Control*, 53(11):2674–2679, 2008.

[18] Y. Nesterov. Smooth minimization of non-smooth functions. *Math. Prog.*, 103(1):127–152, May 2005.

[19] A. B. Poore. Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking. *Comput. Opt. Appl.*, 3(1):27–57, Mar. 1994.

[20] A. M. Rush and M. Collins. A tutorial on dual decomposition and Lagrangian relaxation for inference in natural language processing. *J. Artif. Int. Res.*, 45(1):305–362, Sept. 2012.

[21] D. Sontag, A. Globerson, and T. Jaakkola. Introduction to dual decomposition for inference. In *Optimization for Machine Learning*. MIT Press, 2011.

[22] B. Taskar, S. Lacoste-Julien, and M. I. Jordan. Structured prediction, dual extragradient and Bregman projections. *JMLR*, 7:1627–1653, 2006.

[23] C. J. Veenman, M. J. T. Reinders, and E. Backer. Resolving motion correspondence for densely moving points. *TPAMI*, 23(1):54–72, 2001.

[24] A. R. Zamir, A. Dehghan, and M. Shah. GMCP-Tracker: global multi-object tracking using generalized minimum clique graphs. In *ECCV*, 2012.

[25] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.

---

[9]The source code of the implementation will be made available.