Toward Optimal Matching for 3D Reconstruction of Brachytherapy Seeds*

Christian Labat[1], Ameet Jain[1,3], Gabor Fichtinger[1], and Jerry Prince[2]

[1] Department of Computer Science
[2] Department of Electrical and Computer Engineering,
Johns Hopkins University, Baltimore, MD 21218, USA
[3] Philips Research North America, Briarcliff, NY

Abstract. X-ray C-arm fluoroscopy is a natural choice for intra-operative seed localization in prostate brachytherapy. Resolving the correspondence of seeds in the projection images can be modeled as an assignment problem that is NP-hard. Our approach rests on the practical observation that the optimal solution has almost zero cost if the pose of the C-arm is known accurately. This allowed us to to derive an equivalent problem of reduced dimensionality that, with linear programming, can be solved efficiently in polynomial time. Additionally, our method demonstrates significantly increased robustness to C-arm pose errors when compared to the prior art. Because under actual clinical circumstances it is exceedingly difficult to track the C-arm, easing on this constraint has additional practical utility.

1 Introduction

Intraoperative dosimetric quality assurance in prostate brachytherapy critically depends on discerning the three-dimensional (3D) locations of implanted seeds. The ability to reconstruct the implanted seeds intraoperatively will allow us to make immediate provisions for dosimetric deviations from the optimal implant plan. A method for seed reconstruction from pre-segmented C-arm fluoroscopy images has been proposed, among other works, by Jain et al. in [1], where the 3D coordinates of the implanted seeds are calculated upon resolving the matching of seeds in multiple X-ray images.

At least three images are necessary to eliminate ambiguities. The resulting optimization problem is NP-hard. Heuristic approaches, such as of Jain et al. [1], have been proposed to approximately solve the optimization problem. However, the use of a heuristics leads to *algorithmic error*, in addition to physical errors like the inaccuracy in knowing the relative poses of the C-arm shots (pose error). To tackle this issue, we propose to consider all the images simultaneously, instead of suboptimal subsets of two images such as proposed in [1].

The optimization problem has a salient feature: since the images represent a real situation (i.e. an existing object, the set of seeds, is being imaged), the

---

optimal solution has a near-zero cost when the pose error is low, and is actually zero without pose error. We propose to utilize this feature of the problem to reduce its number of variables, thereby allowing to obtain the optimal solution at a reasonable computational cost. This exact *dimensionality reduction* is only possible when the pose error is sufficiently low. We claim that this is not restrictive in our framework since a high pose error leads to high error in the estimation of the 3D coordinates of the implanted seeds, which is not acceptable. Actually, the idea of dimensionality reduction is not new. For instance, in [1, p. 3480] the original tripartite matching is projected into inspired bipartite matchings, while introducing inaccuracy. In contrast, the proposed method performs dimensionality reduction while ensuring equivalency to the original problem.

The MARSHAL method of Jain et al. has demonstrated solid performance [1] and we chose this work as the benchmark and basis of comparison for our work. A comparison between MARSHAL and our method was conducted to evaluate the sensitivity to pose errors on simulated and phantom data. The proposed method shows significant increase in robustness to pose errors.

2   Method

Consider a collection of X-ray images of a constellation of implanted seeds. We assume that the 2D seed locations can be identified on each of the X-ray images, and we consider the problem of identifying corresponding seeds in all the images. Given these matched seed locations, a reconstruction of the seed locations in 3D can be achieved provided there are no ambiguities. It is more likely that such ambiguities are avoided when there are more X-ray images, but this in turn increases the complexity of the problem. Here we do not consider CT or limited angle tomosynthesis, as our work focuses on reconstruction from a very limited number of images. We propose a solution for three images, which is often sufficient in practice, and which is extendable to more images.

2.1   3D Reconstruction as a Matching Problem

The 3D locations of the implanted seeds, modeled as points, can be reconstructed through 3D triangulation from the X-ray images upon resolving the correspondence of seeds, which is the focus of this paper. Let n be the number of points in the clinical work volume. Let s_{lm} be the position of lth point in m^{th} image. When three images are used, the matching problem can be formulated as an axial 3D assignment problem (3DAP) [1]:

$$\min_{x_{ijk}} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{ijk} x_{ijk}, \quad \text{where} \tag{1}$$

$$\begin{cases} x_{ijk} \in \{0,1\} \\ \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ijk} = 1, & \forall k \\ \sum_{i=1}^{n} \sum_{k=1}^{n} x_{ijk} = 1, & \forall j \\ \sum_{j=1}^{n} \sum_{k=1}^{n} x_{ijk} = 1, & \forall i \end{cases} \tag{2}$$

c_{ijk} is the the cost of matching point s_{i1} to points s_{j2} and s_{k3}. x_{ijk} is a binary variable deciding the correctness of the match $\langle i, j, k \rangle$. (2) force every segmented point to be chosen once. Thus, x represents any feasible global match, with the cost of that correspondence given by $\sum\sum\sum c_{ijk}x_{ijk}$. One good choice for a cost-metric c is projection error (PE) [1]. For any given set of poses and correspondence, the intersection of the three lines that join each projection to its respective X-ray source can be computed using a closed form solution that minimizes the L_2 norm of the error. PE can be computed by projecting this 3D point in each image and then measuring the distance between the projected location and the observed location of the point.

A feasible solution x of the above problem is a 3D permutation array. This problem has $(n!)^2$ feasible solutions. Branch and bound is a classical algorithm for optimally solving the 3DAP. This can be generally achieve only for n small because of the combinatorial explosion. Thus, it has been proposed heuristics that approximately solve the 3DAP, such as MARSHAL in [1]. MARSHAL suboptimally projects the original 3DAP into three distinct 2DAP that can be solved in polynomial time by using the Hungarian algorithm.

We point out that the 3DAP has a salient feature that we can exploit. Since the images represent a real situation, the optimal solution has a near-zero cost when the pose error is low and the optimal cost is actually zero without no pose error. In the next section, we use this feature to reduce the number of variables in the problem, thus permitting us to get the optimal solution at a reasonable computational cost. This new method tackles the complete optimization problem without using suboptimal projections, such as in MARSHAL.

2.2   Dimensionality Reduction

Let $N = n^3$. We rewrite the variables x_{ijk} and the costs c_{ijk} in a vectorial form such that $\boldsymbol{x}, \boldsymbol{c} \in \mathbb{R}^N$. In the sequel we also make use of the notation u_ℓ to denote u_{ijk}. The 3DAP (1)-(2) reads as the following integer linear program

$$P: \quad \min_{\boldsymbol{x} \in \mathcal{C}} \boldsymbol{c}^{\mathrm{t}}\boldsymbol{x}, \tag{3}$$

with the constraint set $\mathcal{C} = \{\boldsymbol{x} : \mathbf{M}\boldsymbol{x} = [1, \ldots, 1]^{\mathrm{t}}, x_\ell \in \{0,1\}\}$, where $\mathbf{M}\boldsymbol{x} = [1, \ldots, 1]^{\mathrm{t}}$ is a matrix form of (2).

Principle. Since the coefficients of x are either 0 or 1 and there must be n 1's, an optimal solution of P can be thought of as the selection of n cost coefficients such that the resulting cost is minimum while constraints $\mathcal{C}$ are satisfied. Given a feasible solution, Lemma 1 (below) states that all cost coefficients that are greater than the cost associated with this solution cannot be selected in the optimal solution. Since those coefficients can never be selected, the dimension of the problem can be reduced by removing those coefficients from further consideration. This yields an equivalent problem of reduced dimensionality. If the reduction in dimensionality is sufficiently large, then the new problem can be solved exactly in reasonable time even though the original problem is far too costly to solve.

Lemma 1. *Let us assume that the cost coefficients c_ℓ are positive. Let $\boldsymbol{x}_0 \in \mathcal{C}$ be a feasible solution. The integer linear problem P defined by (3) is equivalent to the following integer linear problem (ie., they share the same optimal solutions)*

$$P' : \quad \min_{\boldsymbol{x} \in \mathcal{C}} (\boldsymbol{c}')^{\mathrm{t}} \boldsymbol{x}, \quad \text{where}$$

$$c'_\ell = \begin{cases} c_\ell, & \text{if } c_\ell \leq m_P(\boldsymbol{x}_0) \\ \infty, & \text{if } c_\ell > m_P(\boldsymbol{x}_0) \end{cases} \tag{4}$$

and where $m_P(\boldsymbol{x}) = \boldsymbol{c}^{\mathrm{t}} \boldsymbol{x}$ is the cost of problem P at the feasible solution $\boldsymbol{x}$.

Proof. Let $\boldsymbol{x}^* \in \mathcal{C}$ be an optimal solution to problem P. We have for all $\boldsymbol{x}_0 \in \mathcal{C}$

$$m_P(\boldsymbol{x}^*) \leq m_P(\boldsymbol{x}_0). \tag{5}$$

Let us consider $c_\ell > m_P(\boldsymbol{x}_0)$. From (5) we have $c_\ell > m_P(\boldsymbol{x}^*)$. Since c_ℓ are positive and x_ℓ^* are binary, we have necessarily $x_\ell^* = 0$. $\qquad\square$

The dimensionality reduction can be illustrated as follows:

$$P : \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix} \longrightarrow P' : \begin{bmatrix} c'_1 \\ c'_2 \\ \vdots \\ c'_N \end{bmatrix} = \begin{bmatrix} c_{i1} \\ \infty \\ c_{i2} \\ \vdots \\ c_{iK} \end{bmatrix} \longrightarrow \tilde{P} : \begin{bmatrix} \tilde{c}_1 \\ \tilde{c}_2 \\ \vdots \\ \tilde{c}_K \end{bmatrix} = \begin{bmatrix} c_{i1} \\ c_{i2} \\ \vdots \\ c_{iK} \end{bmatrix}$$

The original problem P is equivalent to the reduced problem

$$\tilde{P} : \quad \min_{\tilde{\boldsymbol{x}} \in \tilde{\mathcal{C}}} \tilde{\boldsymbol{c}}^{\mathrm{t}} \tilde{\boldsymbol{x}},$$

where $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{c}} \in \mathbb{R}^K$ ($K \leq N$) and with the constraint set $\tilde{\mathcal{C}} = \{\tilde{\boldsymbol{x}} : \tilde{\mathbf{M}} \tilde{\boldsymbol{x}} = [1, \ldots, 1]^{\mathrm{t}}, \tilde{x}_\ell \in \{0, 1\}\}$ with $\tilde{\mathbf{M}} = \mathbf{MR}$ and where $\mathbf{R}$ is the *dimensionality reduction matrix* of size $N \times K$ such that $\begin{bmatrix} x_{i1} & 0 & x_{i2} & 0 & \ldots & x_{iK} \end{bmatrix}^{\mathrm{t}} = \mathbf{R} \begin{bmatrix} \tilde{x}_1 & \tilde{x}_2 & \ldots & \tilde{x}_K \end{bmatrix}^{\mathrm{t}}$. Once the reduced problem $\tilde{P}$ is solved, the optimal solution to the original problem P is simply given by $\boldsymbol{x}^* = \mathbf{R} \tilde{\boldsymbol{x}}^*$.

Note that a dimensionality reduction ($K < N$) is not guaranteed for the general 3DAP, even in the most favorable case $m_P(\boldsymbol{x}_0) = m_P(\boldsymbol{x}^*)$. However, a dimensionality reduction occurs in the case of our problem since the range of the cost coefficients is wide and the optimal solution has a near-zero cost when the pose error is low. The practical interest clearly depends on the dimensionality reduction ratio (K/N). We show next that this ratio actually can be improved.

Improving dimensionality reduction. Lemma 1 uses only the integer constraint in $\mathcal{C}$ to reach dimensionality reduction. But using the fact that the feasible set $\mathcal{C}$ comprises permutation matrices, is is possible to reduce the dimensionality of this problem even further. To demonstrate this, we start with the following Lemma.

Lemma 2. *The integer linear problem P defined by (3) is equivalent to the following integer linear problem*

$$P'' : \quad \min_{\boldsymbol{x} \in \mathcal{C}} (\boldsymbol{c}'')^{\mathrm{t}} \boldsymbol{x}$$

where the minimum cost in each row is subtracted from the entire row

$$c''_{ijk} = c_{ijk} - \min_{j,k} c_{ijk}. \tag{6}$$

Proof. For lack of space, the proof is not detailed.

To show that this Lemma permits further dimensionality reduction, let us apply Lemma 1 on the new problem P''. From (4), the cost coefficient c''_ℓ is equivalent to ∞ if $c''_\ell > m_{P''}(\boldsymbol{x})$. According to (6) the former condition is equivalent to

$$c_{ijk} > m_P(\boldsymbol{x}) - \left(\sum_{i=1}^{n} \min_{j,k} c_{ijk} - \min_{j,k} c_{ijk} \right)$$

The latter condition is less restrictive than $c_\ell > m_P(\boldsymbol{x})$ since the cost coefficients are positive. As a result, the dimensionality reduction is higher within the new problem P'' than within the original problem P. It is then preferable to consider problem P'' instead of problem P since Lemma 2 ensures that they are equivalent. It is actually possible to reduce dimensionality even further. The operation (6) can also be performed successively for the columns and depths to decrease the cost coefficients, while still ensuring equivalency to the original problem.

2.3   Optimization Strategy

Integer Programming / Linear Programming. The Integer Program (1)-(2) can be directly solved with standard techniques such as branch and bound. IP problems are NP hard, however, and may take an exponential amount of computational time. It has been shown that the linear program corresponding to the 2D assignment problem (2DAP) has an integer solution even without integer constraints [2]. As well, this linear program can be solved efficiently in polynomial time using interior point methods for instance [3]. To our knowledge, however, there is no analogous result for the 3DAP. Nevertheless, we have gone ahead and implemented the linear program for the 3DAP problem followed by a test to see if its solution is binary (up to numerical errors). In all of our experiments, we have never obtained a nonbinary solution to this problem, which points to the potential validity of the 2D theoretical result in 3D as well. (We are currently investigating this theoretical issue.)

Dimensionality reduction thresholding. From Lemma 1, the degree of dimensionality reduction depends on the cost $m_P(\boldsymbol{x}_0)$ of a feasible solution $\boldsymbol{x}_0$. It is possible to use a suboptimal algorithm, such as MARSHAL, to find a feasible solution $\boldsymbol{x}_0$. Unfortunately, when there is high pose error, MARSHAL often provides such a suboptimal solution that very little dimensionality reduction can be

be achieved. Therefore, we propose instead to choose a threshold parameter η, which is essentially a "guess" as to what the cost of a feasible solution might be. This permits us to reduce the dimensionality of the problem and run the linear problem on the resultant problem. If the solution of this problem is integer and it has a cost lower than η, then it must be optimal. If the resultant cost is larger than η then the solution might be optimal, but we cannot guarantee it. It is then our option whether to accept a (potentially) suboptimal solution or to increase η and rerun the linear problem until we have a guaranteed optimal solution.

In our experiments, we determine η in the following way. Rank order all costs from lowest to highest and pick an integer K. Let η be the value of the Kth smallest cost coefficient. The influence of K on the rate of feasibility and optimality of the proposed method is experimentally studied in the next section.

3   Simulation and Phantom Experiments

We present a comparison between the MARSHAL algorithm and the proposed method using simulation and phantom data. We got a copy of the MARSHAL code from the authors for comparison [1]. Both algorithms were implemented in Matlab 7.1 on a Linux PC (Intel EM64T 3.6 GHz, 24GB RAM).

3.1   Evaluation of Pose Errors Sensitivity

Two separate comparisons were performed. One compared the two algorithms to translational errors and the other to rotational errors. These are both common errors in C-arm position calibration in the operating room. Random error was modeled using a uniform probability density function. When we report results for an h error this means that each of the three components of error (in either translation or rotation) were generated as independent random variables uniformly distributed on $[-h, h]$. Following [1], a statistical bias in translation was incorporated in the generation of the datasets to account for the expected differences in directional errors in fluoroscope tracking using a fiducial. In particular, we assumed that the in-plane error is a factor of five times smaller than the through-plane error h. No analogous bias was used in the rotation errors.

Realistic simulations of prostate brachytherapy seeds implants were generated with seeds density of 2 and 2.5 seeds/cc and prostate size of 35 and 45 cc. The number of seeds in the implants were $n = \{72, 84, 96, 112\}$. A cone angle of 10° was used for the acquisition of the three simulated X-ray images. Averaged results from a total of $2,000$ datasets are shown in Fig. 1.

The proposed method, with $100n$ cost coefficients remaining after dimensionality reduction, performs significantly better than MARSHAL, as shown in Fig. 1(a)-(d). It still matches correctly 89% of the seeds when rotation error reaches 4°, while MARSHAL drops to 59%. The proposed method still matches correctly 99% of the seeds when the translation error reaches 10 mm, while MARSHAL drops to 72%. The computational time of the proposed method is higher than that of MARSHAL as shown in Fig. 1(e)-(f). We point out that most of the computational time of the proposed method (solid line) is actually used in the
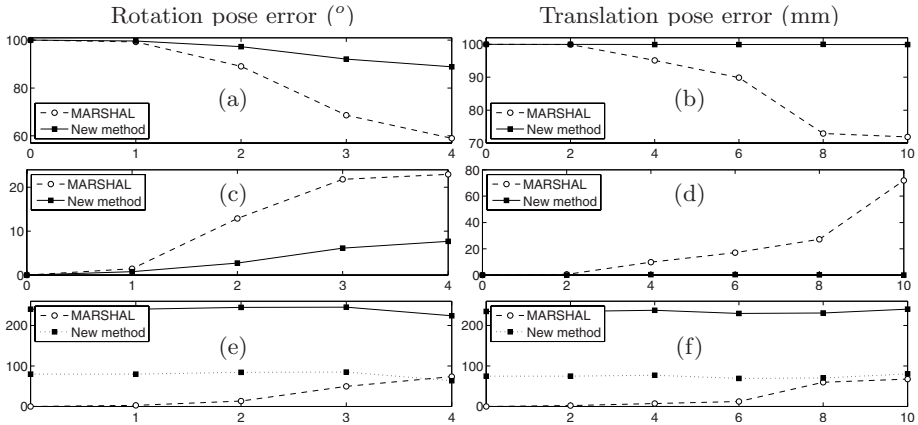
Fig. 1. Performance comparison between MARSHAL and the new method for pose error. (a)-(b): Mean of matching rate (%), (c)-(d): STD (%), (e)-(f): Computation time (s), dotted line: time required solely for cost minimization (new method).

Table 1. Feasibility and optimality of the proposed method for pose error

Number of cost coef.	Feasibility rate					Guaranteed optimality rate				
	20n	50n	100n	500n	1000n	20n	50n	100n	500n	1000n
Error 1 ^{o}	0.9	0.96	1	1	0.98	0.88	0.91	0.94	0.98	1
Error 2 ^{o}	0.9	0.98	1	0.98	0.94	0.35	0.38	0.46	0.6	0.71
Error 3 ^{o}	0.58	0.94	0.98	0.94	0.92	0.1	0.13	0.13	0.22	0.27
Error 4 ^{o}	0.42	0.83	0.92	0.92	0.85	0.15	0.1	0.07	0.07	0.1
Error 2 mm	0.96	1	1	1	0.94	1	1	1	1	1
Error 4 mm	0.94	0.98	1	1	0.92	0.56	0.6	0.65	0.77	0.89
Error 6 mm	0.96	0.96	1	0.96	0.88	0.35	0.41	0.46	0.63	0.64
Error 8 mm	0.77	0.9	0.98	0.94	0.88	0.14	0.19	0.19	0.31	0.38
Error 10 mm	0.58	0.92	0.96	0.94	0.77	0	0.02	0.04	0.07	0.08

computation of the cost coefficients. However, computing all the cost coefficients is not required since most of them will eventually be thrown out by dimensionality reduction. After further code optimization, we expect the computation time to reduce, near to the time required solely for cost minimization (dotted line).

The feasibility and optimality rates of the proposed method as a function of the number of remaining cost coefficients after dimensionality reduction are shown in Tab. 1. It is expected that the feasibility rate should increase as a function of the number of cost coefficients. This is true for smaller numbers of cost coefficients but, surprisingly, the feasibility rate decreases when the number of cost coefficients reaches $500n$ ($\simeq 50,000$). This is due to the failure of `linprog` in Matlab using default parameters because "one or more of the residuals, duality gap, or total relative error has stalled". These cases were not displayed in Fig. 1, and we are currently investigating how to cope with them.

The guaranteed optimality rate increases as a function of the number of cost coefficients. For low errors (1° and 2 mm), all solutions are guaranteed optimal given an acceptable dimensionality ratio. We point out that the guaranteed

optimality is only a sufficient condition. As shown in Fig. 1(b), all solutions from 0 to 8 mm translation error of the proposed method correspond to perfect matching (100%), even if they are not all guaranteed optimal.

3.2   Phantom Experiments

A radiographic fiducial was used to track the C-arm (0.56 mm translation; 0.33^o rotation accuracy) and was accurately attached to a random point cloud phantom. Phantoms of $\{40, 55, 70, 85, 100\}$ points with 1.56 $points/cc$ were used. Six X-ray images within a 20° cone around the AP-axis were *randomly* taken using an *Philips Integris V3000* fluoroscope and dewarped. Thus both the seed locations and X-ray pose were not biased/optimized in any way, closely representing an uncontrolled surgical scenario. Each image was hand segmented to establish the true segmentation and correspondence.

Both MARSHAL and the proposed method perform very well on phantoms, achieving almost perfect matching, as shown in Tab. 2. Note that the accuracy of the radiographic fiducial ensures here a low pose error. The proposed method is significantly slower compared to MARSHAL. We point out that the proposed method uses here more than 90% of the computational time for the n^3 cost coefficients. We expect to reduce significantly this time as explained in Sect. 3.1.

Table 2. Performance of MARSHAL and the proposed method on phantoms

	MARSHAL					Proposed method				
Number of seeds	40	55	70	85	100	40	55	70	85	100
Mean Match. (%)	97.6	100	98	97.7	98.2	98	99.4	97.1	100	98
STD Match. (%)	3.6	0	2.3	3.2	2.3	2.6	1.4	0	0	0
Time (s)	0.3	0.6	1	2.5	3.1	12.6	32	64.6	106.6	185

Conclusion and Future Work. In summary, we achieved significant increase in the robustness to pose errors compared to [1] by considering all images simultaneously, instead of using subsets. Experimentally, our method ensured optimality for small pose errors. C-arm tracking is a cumbersome process and easing on this constraint has great practical utility. With our method, a less accurate estimation of the pose may suffice. For example in [4], starting from an initial guess, pose was further estimated iteratively using the current 3D reconstruction, yet the seed matching remained susceptible to pose errors. Applying our method to [4] promises a clinically viable solution without using external tracker or encoder on the C-arm. We are also extending the method to reconstructing overlapping seeds that are occluded in one or more X-ray images [5,6].

References

1. Jain, A., et al.: Matching and reconstruction of brachytherapy seeds using the hungarian algorithm (marshal). Med. Phys. 32(11), 3475–3492 (2005)
2. Papadimitriou, C.H., Steiglitz, K.: Combinatorial optimization: algorithms and complexity. Prentice-Hall, Inc., Englewood Cliffs (1982)

3. Bertsekas, D.: Nonlinear Programming. Athena Scientific, Belmont, USA (1999)
4. Jain, A., et al.: C-arm tracking and reconstruction without an external tracker. In: Larsen, R., Nielsen, M., Sporring, J. (eds.) MICCAI 2006. LNCS, vol. 4190, pp. 494–502. Springer, Heidelberg (2006)
5. Su, Y., et al.: Prostate brachytherapy seed localization by analysis of multiple projections: Identifying and addressing the seed overlap problem. Med. Phys. 31(5), 1277–1287 (2004)
6. Narayanan, S., et al.: Three-dimensional seed reconstruction from an incomplete data set for prostate brachytherapy. Phys. in Med. and Bio. 49(15), 3483–3494 (2004)