Optimized Cranial Bandeau Remodeling *

James Drake\textsuperscript{2}, Marina Drygala\textsuperscript{1}, Ricardo Fukasawa\textsuperscript{1}, Jochen Koenemann\textsuperscript{1}, Andre Linhares\textsuperscript{1}, Thomas Looi\textsuperscript{2}, John Phillips\textsuperscript{2}, David Qian\textsuperscript{1}, Nikoo Saber\textsuperscript{3}, Justin Toth\textsuperscript{1}, Chris Woodbeck\textsuperscript{1}, and Jessie Yeung\textsuperscript{1}

University of Waterloo, Waterloo, Ontario; The Hospital for Sick Children, Toronto, Ontario; MyJiko, Toronto, Ontario

Abstract. Craniosynostosis, a condition affecting 1 in 2000 infants, is caused by premature fusing of cranial vault sutures, and manifests itself in abnormal skull growth patterns. Left untreated, the condition may lead to severe developmental impairment. Standard practice is to apply corrective cranial bandeau remodeling surgery in the first year of the infants life. The most frequent type of surgery involves the removal of the so-called fronto-orbital bar from the patients forehead and the cutting of well-placed incisions to reshape the skull in order to obtain the desired result.

In this paper, we propose a precise optimization model for the above cranial bandeau remodeling problem and its variants. We have developed efficient algorithms that solve best incision placement, and show hardness for more general cases in the class. To the best of our knowledge this paper is the first to introduce optimization models for craniofacial surgery applications.

Keywords: Craniosynostosis, dynamic programming, healthcare

\* The authors were supported by the NSERC Discovery Grant program and acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC).
1 Introduction

While adult skulls are a single solid piece of bone, infant skulls are in fact composed of several bone pieces that start to fuse at the joints (cranial vault sutures) as the infant grows. However, when these sutures fuse prematurely, it leads to a condition called craniosynostosis.

The purpose of this work is to study optimization models arising from the treatment of craniosynostosis, which affects 1 in 2000 infants [17]. Craniosynostosis often yields abnormal growth patterns of the infants skull, ultimately resulting in severe deformations (see Figure 1.(a)). Left untreated, it may also lead to increased intracranial pressure, and this in turn may cause stunted mental growth, vision impairment and several other severe impairments of the patient.

![Fig. 1: (a) shows the skull of a craniosynostosis patient [18], (b) the schematics of a bandeau [16], and (c) a bandeau attached to metal template [9]](image)

To better understand the optimization problem that arises, we first focus on one of the methods to treat the condition. Surgical treatment of craniosynostosis, so-called bandeau remodeling surgery, is often applied within the first year of an infants life [14]. A central component in such surgeries is the reshaping of the so-called fronto-orbital bar, or bandeau, which is a strip of bone from the patients forehead above the eyebrows (see Figure 1.(b)). The surgeon removes the bandeau from the patients skull, then performs several well-placed incisions into this piece of bone in order to make it more malleable. Mechanical linkages are inserted at the incision points and the front-orbital bar is now shaped by attaching it to a metal template that has the desired ideal skull shape (see Figure 1.(c)). The reshaped bandeau is then inserted back into the patients forehead.

One important part of such problems is the determination of the ideal skull shape. Recently, a team of surgeons and researchers at the Hospital for Sick Children developed a library of normative pediatric skulls modelling ideal skull shapes of infants of varying ages [16]. Based on these skulls, the hospital now has such steel skull templates and the surgeon chooses the template that is most appropriate to each individual craniosynostosis patient, based on the patients age and size. The availability of such individual steel templates during operation has led to a more standardized, objective and precise correction of craniosynostosis [16].

A key factor in adequately approximating the ideal skull shape is to strategically place the kerf points on the deformed bandeau. This important part of the surgery is currently performed based on ad hoc methods, using the surgeons experience, intuition and artistic talent. Obviously, this means that the quality of the outcome of this procedure depends integrally on the surgeons experience and skill. Another major drawback is the current difficulty in training new surgeons, and ensuring standardized and consistent surgical outcomes.

In this paper we focus on the surgeons task of finding locations for the fronto-orbital bar incisions, the bandeau reshaping problem (BRP). We define a formal mathematical problem, the curve refitting problem (CR), which models this task and allows us to evaluate the quality of an operation’s outcome. We present an efficient algorithm for this problem, and show hardness results for some of its variants. To the best of

---

1 Surgeons do not cut the bandeau, but rather 'kerf’ [1] it at prescribed cut locations to make it bendable.
our knowledge, our work presents the first formal injection of optimization methods into the realm of plastic craniofacial surgery.

1.1 Related Work

The use of optimization in healthcare has received increased attention in recent years. The range of applications has gone from staff and facilities scheduling (e.g. [2,3]) and outpatient scheduling [4] to cancer radiation treatment [11,7]. However, to the best of our knowledge, this is the first work that applies an optimization approach to a problem that deals with bone reconstruction. Indeed, most of the quantitative work in craniosynostosis treatment has focused on measuring the severity of cases [15,6].

There have also been some similar problems studied in different contexts. For instance, [12] looks at the problem of reconstructing an artifact based on its fragments. However, such work is based on reconstructing a previously known artifact based on how similar the edges of the broken pieces are, which does not apply in our case. In addition, there are works on computational geometry that study how to best match two sets of points, like the iterative closest point (ICP) problem [5] and the Procrustes problem. [8] However these problems only allow some very well-defined uniform transformations (e.g. scaling, rotating) to be applied to the set of points and thus cannot be applied in our setting, where one seeks to maintain some parts of the skull intact, while transforming (rotating) other parts, which are obtained after the incisions are done, independently from the remainder of the skull.

Therefore, none of the previous literature found would apply in solving the desired (BRP) and thus, there is a need to develop new methodology to solve this problem.

1.2 Our Contributions and Outline

We consider the main contribution of our work to be the introduction of formal discrete optimization methods for craniofacial plastic surgery. Moreover, we believe that we are the first to provide a formal abstract model (CR) of the craniofacial surgical problem of reshaping the fronto-orbital bar, which is presented in Section 2. We then present an exact algorithm for the problem based on dynamic programming in Section 3, along with the presentation of an efficient implementation of the algorithm and empirical evaluation in Section 4. Lastly, we illustrate in Section 5 a variant of the problem which is computationally hard.

2 Computational Models

2.1 Curve Reshaping Problem

In a Curve Reshaping Problem (CR) we are given, as inputs, two piecewise linear continuous functions (henceforth referred to as curves) of the form \( f : [0, s] \rightarrow \mathbb{R} \) and \( g : [0, t] \rightarrow \mathbb{R} \), each presented as input using one of two models: the implicit model or the explicit discretization model, see Section 2.2. The choice of model has an impact on the complexity of the problem. The curves are used to represent bandeaus from a top-down perspective, with curve \( f \) for the deformed bandeau and curve \( g \) for the ideal bandeau.

We are also given as input to (CR) a parameter \( k \in \mathbb{Z}_{\geq 1} \), indicating the number of cuts on \( f \) we intend to make. The last piece of input we are given is a function \( c : 2^{[0, s]} \times [0, t] \rightarrow \mathbb{R} \cup \{\infty\} \). The value of \( c([a, b], \ell, r) \) measures the cost of using the segment \( f_{[a, b]}\ell \) of \( f \), in order to cover a segment of \( g \) by clamping the left endpoint of \( f_{[a, b]}\ell \) at \( \ell \) and the right endpoint at \( r \). See subsection 4.1 for an elaboration, where we discuss a particular cost function used in our target application. In our models we treat the function \( c \) as a black box oracle, although in future work it would be interesting to identify natural cost functions and exploit their structure in the design of algorithms.

In a (CR) problem, the decisions are as follows: Choose \( k + 1 \) positions on the deformed curve, \( p_0 < p_1 < \cdots < p_k \in P \), where \( P \subseteq [0, s] \cap Q \) is a finite set of feasible cut positions given by the discretization (see subsection 2.2) of \( f \). When \( p_0 > 0 \) or \( p_k < s \) part of the ends of the deformed curve are discarded. These cuts segment \( f \) into pieces \( f_{[a, b]}\ell \) which we will map onto the ideal curve \( g \).

To map the segments of \( f \) onto \( g \) we make the following decisions: for each \( i \in [k] \), choose two positions \( \ell_i, r_i \in Q \) indicating positions of the left and right endpoints that the \( i \)th segment of \( f \) will take on the ideal curve. Here \( Q \subseteq [0, t] \cap Q \) is a finite set of feasible clamping positions given by the discretization of \( g \).
The placement of our segments of \( f \) on the ideal curve \( g \) cannot be allowed to overlap, except at their endpoints. Formally this means that for all \( i \neq j \in [k] \),

\[
| (\text{cov}(\ell_i, r_i) \cap \text{cov}(\ell_j, r_j)) \cap Q | \leq 1,
\]

where \( \text{cov}(\ell_i, r_i) := [\min(\ell_i, r_i), \max(\ell_i, r_i)] \).

The objective in our (CR) problem is to minimize the total dissimilarity induced by our solution. To penalize discarding portions of the deformed curve, we select a parameter \( \gamma > 0 \) and add the term \( \gamma (p_0 + (s - p_k)) \) to our objective. In summary, we may formally state our problem as follows:

\[
\min \sum_{i=1}^{k} c([p_{i-1}, p_i], \ell_i, r_i) + \gamma (p_0 + (s - p_k)) \tag{CR}
\]

s.t. \( | (\text{cov}(\ell_i, r_i) \cap \text{cov}(\ell_j, r_j)) \cap Q | \leq 1 \quad \forall i \neq j \in [k] \)

\[
p_0 < p_1 < \cdots < p_n \in P
\]

\[
\ell, r \in Q^k.
\]

One can generalize the (CR) problem assumptions by replacing our assumptions about uncoverage and overlap with general penalty functions for: leaving a piece of the ideal curve uncovered, leaving a piece of the deformed curve unused, and having overlap in how your segments of the deformed curve cover the ideal curve. We leave these generalizations to future work.

### 2.2 Implicit and Explicit Discretized Curves

**Implicit** When presenting a curve \( \eta : [0, u] \rightarrow \mathbb{R} \) via an implicit discretization, we give, as input to the algorithm, the value \( u \) bounding the domain of \( \eta \). We also give parameter \( \delta > 0 \) as input. We construct a sequence of points

\[
a_0, a_1, \ldots, a_{\lceil u/\delta \rceil}
\]

such that \( a_0 = 0, a_{\lceil u/\delta \rceil} = s \) and for all \( i \in \lceil u/\delta \rceil \), \( a_i = a_{i-1} + \delta \). If \( \eta \) is the deformed curve \( f \) then set

\[
P = \{a_0, a_1, \ldots, a_{\lceil s/\delta \rceil}\}.
\]

If instead we are presenting a discretization of the ideal curve \( g \) then set

\[
Q = \{a_0, a_1, \ldots, a_{\lceil s/\delta \rceil}\}.
\]

**Explicit** When presenting the curve \( \eta : [0, u] \rightarrow \mathbb{R} \) via an explicit discretization, we give a set of points \( D \subseteq [0, u] \) to the algorithm as input, such that \( \eta \) is a piecewise linear interpolation of

\[
\{(x, \eta(x)) : x \in D\}.
\]

If \( \eta \) is the deformed curve \( f \) then set \( P = D \). If \( \eta \) is the ideal curve \( g \) then set \( Q = D \).

In either case, we assume an algorithm for our (CR) has access to an oracle who can evaluate the value of \( f(x) \) (or \( g \) respectively) for any \( x \in P \) (\( x \in Q \) respectively) in polynomial time.

### 2.3 Without Rearrangement

In a general 3-dimensional cranial vault remodeling surgery, where the entire skull is being reorganized, it is possible to take pieces and rearrange their relative positions. The (CR) model as described previously also admits this possibility. In practice, the (BRP) surgery for reshaping the bandeau does not admit this. That is to say, the relative positions of the cut segments of the deformed curve are not changed by their mapping onto the ideal curve.

Mathematically we can model this by adding constraints saying that for all \( i \in [k] \setminus \{1\} \), \( r_{i-1} = \ell_i \), and adding constraints for each \( i \in [k] \) saying that \( \ell_i \leq r_i \). We refer to the resulting problem as a (CR) problem without rearrangement.
3 Dynamic Programming Algorithm

In this section, we present a dynamic programming algorithm a particular class of (CR) problems. For the remainder of this section, we consider an instance \( I \) of the (CR) problem without rearrangement, where the deformed curve \( f : [0, s] \rightarrow \mathbb{R} \) and the ideal curve \( g : [0, t] \rightarrow \mathbb{R} \) are explicitly discretized. Let \( c \) be the associated cost function, and let \( k \in \mathbb{Z}_{\geq 1} \) be the associated number of cuts.

For any \( [a, b] \subseteq [0, s] \) and any \( [d, e] \subseteq [0, t] \), and \( k' \in [1, k] \cap \mathbb{Z} \), let

\[
CR([a, b], [c, d], k')
\]
denote the optimal value of the (CR) problem without rearrangement using cost function \( c \), overlap parameter \( \gamma = 0 \), with \( k' \) cuts, on the deformed curve \( f|_{[a, b]} \) presented via an explicit discretization with feasible cut positions \( P \cap [a, b] \) and the ideal curve \( g|_{[d, e]} \) presented via an explicit discretization with feasible clamp positions \( Q \cap [d, e] \).

Our goal is to give a recursive formula for \( CR([a, b], [d, e], k') \) that can be used to design a dynamic program.

**Theorem 1.** For any \( [a, b] \subseteq [0, s] \) and any \( [d, e] \subseteq [0, t] \), and \( k' \in [1, k] \cap \mathbb{Z} \) the following recursive equation holds

\[
CR([a, b], [d, e], k') = \begin{cases} 
\min_{(p, q) \in P \cap [a, b] \times Q \cap [d, e]} (c([a, p], d, q) + CR([p, b], [q, e], k' - 1)), & \text{if } k' > 1 \\
c([a, b], d, e), & \text{otherwise}
\end{cases}
\]

**Proof.** We proceed by induction on \( k \). The case \( k = 1 \) is trivial. Since the entire deformed curve \( f|_{[a, b]} \) is assigned to cover the entire ideal curve, \( g|_{[d, e]} \) the solution pays cost \( c([a, b], d, e) \).

Hence we may assume \( k' > 1 \) and the inductive hypothesis holds on all \( CR(\cdot, \cdot, k'') \) where \( k'' < k' \). Let \( p_1^*, \ldots, p_{k'}^* \in P \cap [a, b] \) and \( (\ell_1^*, r_1^*), \ldots, (\ell_{k'}^*, r_{k'}^*) \in Q^2 \cap [d, e]^2 \) be an optimal solution to \( CR([a, b], [d, e], k') \). Observe that

\[
CR([a, b], [d, e], k') = \sum_{i=1}^{k'} c([p_{i-1}^*, p_i^*], \ell_i^*, r_i^*)
\]

\[
= c([a, p_1^*], d, r_1^*) + \sum_{i=2}^{k'} c([p_{i-1}^*, p_i^*], \ell_i^*, r_i^*) \geq \min_{(p, q) \in P \cap [a, b] \times Q \cap [d, e]} (c([a, p], d, q) + CR([p, b], [q, e], k' - 1))
\]

The underbrace inequality above follows since \( p_1^*, \ldots, p_k^* \) and \( (\ell_2^*, r_2^*), \ldots, (\ell_{k'}^*, r_{k'}^*) \) is a feasible solution to the (CR) problem without rearrangement using \( k' - 1 \) cuts on curves \( f|_{[p_1^*, b]} \) and \( g|_{[\ell_{k'+1}^*, e]} \). Note that since this problem is without rearrangement, \( \ell_1^* < r_1^* \) and \( r_{k'}^* = \ell_{k'}^* \).

By Theorem 1, \( I \) can be solved using standard dynamic programming techniques utilizing a table of size \( O(|P| \cdot |Q| \cdot k) \). Each table entry can be computed in \( O(|P| \cdot |Q|) \) time, yielding an algorithm solving (CR) in time \( O(|P|^2 \cdot |Q|^2 \cdot k) \). This is a pseudo-polynomial time algorithm, but when \( k \) is fixed, this becomes a polynomial time algorithm. This consideration is justified, since the experiments in Section 4 demonstrate that a very small number of cuts are needed relative to the size of \( P \) and \( Q \) in order to obtain an extremely high quality bandeau reconstruction.

For \( \gamma > 0 \) the overlap penalty \( \gamma(p_0 + (s - p_k)) \) is completely determined once \( p_0 \) and \( p_k \) are fixed. Then it remains to solve a (CR) problem with 0 overlap penalty on the deformed curve restricted to \([p_0, p_k]\). By enumerating our choices for \( p_0, p_k \) and solving the subproblems with 0 overlap penalty using Theorem 1, we get the following corollary:

**Corollary 1.** Any (CR) problem without rearrangement can be solved in time polynomial in \(|P|, |Q|\) and linear in \( k \).
4 Computational Experiments

In this section, we describe results obtained by testing our dynamic programming algorithm on genuine cases of craniosynostosis provided by the Hospital for Sick Children, as well as on some extreme geometries tailored to challenge the robustness of our algorithm. In subsection 4.1 we describe a concrete example of a cost function $c$ for our (CR) problem, which can be used to empirically evaluate the closeness of fit between a bandeau and the ideal curve. Then subsection 4.3 gives some notes on our implementation of the algorithm in code. Considering the size of the instances and desired response time by surgeons, many non-trivial optimizations were needed to attain a desirable runtime. Finally, subsection 4.2 presents the findings from our experiments.

4.1 Computing the Dissimilarity

To evaluate the quality of a solution, we define, for any pair of curves $f : [0, s] \to \mathbb{R}_{\geq 0}$ and $g : [0, t] \to \mathbb{R}_{\geq 0}$, a dissimilarity measure $d(f, g) \in \mathbb{R} \cup \{\infty\}$. Note that the dissimilarity measure is not necessarily symmetric, that is, the measures $d(f, g)$ and $d(g, f)$ are not necessarily equal. Before proceeding we reiterate that the algorithm does not rely on any properties of $d$, and hence can be applied to different measures at the discretion of the user.

We will use $L_f$ and $R_f$ as shorthands for the left and right endpoints $(0, f(0))$, and $(s, f(s))$ of $f$, respectively. We say that two curves $f$ and $g$ match if the Euclidean distance between the endpoints of $f$ is approximately equal to the Euclidean distance between the endpoints of $g$. Formally, we introduce a parameter $\alpha \in [0, 1]$, and we say that $f$ and $g$ match if

$$1 - \alpha \leq \frac{|L_g - R_g|}{|L_f - R_f|} \leq 1 + \alpha .$$

If $f$ and $g$ do not match, then we define $d(f, g) = \infty$. Next, we describe how to compute $d(f, g)$ when they do match. First, we rotate $g$ around $L_g$ so that its right endpoint is moved to the point $(x(L_g) + |L_g - R_g|, y(L_g)),$ and we denote by $\tilde{g}$ the curve thus obtained. Next, we modify the curve $f$ as follows. We scale it by a factor

$$\frac{|L_g - R_g|}{|L_f - R_f|},$$

so that the Euclidean distance between its endpoint becomes equal to the Euclidean distance between the endpoints of $\tilde{g}$. Next, we translate and rotate it so that its left and right endpoints are mapped to $L_{\tilde{g}}$ and $R_{\tilde{g}}$ respectively. We denote by $\tilde{f}$ the curve thus obtained. Finally, we define $d(f, g)$ to be the area between the modified curves $f$ and $\tilde{g}$, that is, we let

$$d(f, g) = \int_{x(L_f)}^{x(R_f)} |\tilde{f}(x) - \tilde{g}(x)| \, dx .$$

Figure 2 shows an example of two curves $f$ and $g$ and how $d(f, g)$ is calculated.

Now we are ready to define a cost function $c : 2^{[0, s]} \times [0, t] \times [0, t] \to \mathbb{R} \cup \{\infty\}$ for the (CR) problem. For all $[a, b] \subseteq [0, s]$ and all $\ell, r \in [0, t]$ we define

$$c([a, b], \ell, r) = \begin{cases} d(f|[a,b], g|_{[\ell,r]}), & \text{if } \ell \leq r \\ d(f'|_{[r,a]}, g|_{[\ell,r]}), & \text{otherwise,} \end{cases}$$

where $f'$ is the function defined by $f'(x) = f(-x)$, obtained by reflecting $f$ in the first coordinate axis.

Note that curves presented via an explicit discretization, such as those presented to the DP algorithm implied by Theorem 1 are piecewise linear functions whose inflection points are all part of the input. Thus evaluating $d$ on pairs of such curves can be done in polynomial time (in the number of pieces), and hence $c$ can be queried in polynomial time.

4.2 Results
The dynamic programming algorithm presented in Section 3 was implemented in Python 3.6.7 and Numpy 1.15, and OpenCL 1.2 to support precomputing c (see subsection 4.3). For our tests, the PC running the algorithm had an i7-4770k processor, NVIDIA GTX 780 GPU, with 16Gb DDR3-1866 RAM, with Windows 10 build 1803 operating system.

The Hospital for Sick Children provided some test patient data which we used as a starting point to generate a family of test instances. They gave us the test curves Deformed 1 and 2, and Metopic, Metopic BM, and Metopic CS. We added the remaining test instances. Extreme 1, 2, and 3 represent degenerate geometries which do not necessarily represent a realistic scenario but are interesting for testing how robust the algorithm is. The remaining test cases were generated to simulate realistic curves one could expect in the operating room. The curves are given to the algorithm as an explicit list of points describing a piecewise linear curve. These points were generated from scans so that the distance between two consecutive points are between 0.5 and 1mm. This resulted in instances with usually close to 200 points, with the exception of the Sagittal curves which were on the order of 400 points.

Table 1 presents the data from our computational experiments. Our measure of solution quality is the (expected) post-operation ABCs for different numbers of cuts k. We obtained recommended cut positions when using cuts from values 1 to 13 when measuring the total time to compute, but decided to show ABC for k-values: 3, 5, 7, 9, 11, and 13 in the interest of space.

As can be observed from Table 1, the ABC seems to decay rapidly as the number of cuts increases. This relationship is shown visually by the line graph in Figure 4. This decay suggests that rearrangement is not needed in this application to get good outcomes, providing a principled justification for not using rearrangement in surgical procedures.

| Case          | Pre-ABC | T(s)  | k = 3 | k = 5 | k = 7 | k = 9 | k = 11 | k = 13 |
|---------------|---------|-------|-------|-------|-------|-------|-------|-------|
| Deformed 1    | 709.66  | 0.85  | 90.74 | 66.82 | 59.45 | 49.00 | 44.12 | 44.12 |
| Deformed 2    | 353.53  | 0.85  | 38.83 | 24.92 | 14.06 | 10.29 | 8.68  | 7.36  |
| Extreme 1     | 317.35  | 0.48  | 40.09 | 19.56 | 8.84  | 5.19  | 4.20  | 3.57  |
| Extreme 2     | 212.44  | 1.34  | 37.67 | 21.67 | 7.12  | 4.68  | 3.97  | 3.60  |
| Extreme 3     | 603.29  | 0.74  | 37.35 | 17.05 | 9.14  | 4.93  | 3.24  | 2.82  |
| Metopic       | 819.03  | 0.86  | 73.90 | 55.38 | 45.19 | 38.18 | 31.30 | 25.11 |
| Metopic BM    | 520.08  | 1.19  | 100.84| 71.26 | 46.89 | 31.02 | 25.84 | 21.55 |
| Metopic CS    | 329.05  | 1.03  | 43.90 | 37.22 | 31.91 | 26.57 | 21.48 | 16.91 |
| Metopic Line 1| 223.66  | 0.66  | 49.64 | 16.32 | 6.62  | 4.92  | 3.57  | 2.74  |
| Metopic Line 2| 418.18  | 0.67  | 91.68 | 42.04 | 16.65 | 9.72  | 7.54  | 6.07  |
| Metopic Line 3| 332.26  | 0.67  | 70.97 | 33.95 | 14.93 | 8.67  | 6.96  | 6.25  |
| Sagittal 1    | 317.69  | 2.81  | 41.53 | 17.38 | 9.10  | 6.32  | 5.11  | 4.53  |
| Sagittal 2    | 337.73  | 3.45  | 28.32 | 15.45 | 9.54  | 6.23  | 5.37  | 4.65  |
| Sagittal 3    | 407.55  | 3.49  | 68.09 | 23.00 | 12.80 | 8.98  | 7.07  | 5.87  |

Table 1: Computational Results - ABC for a progression of cuts

We note that the computational times are very low, even in the extreme cases, and thus we consider that for all practical purposes, the running times are acceptable from the application point of view. Moreover,
Figure 3 gives a qualitative idea what the ABC measures represent. In that figure, the green line represents the ideal bandeau, the red line represents the bandeau before surgery, with the corresponding cuts, and the blue line represents the expected final outcome of the surgery. As can be seen, the result that the algorithm outputs seems to be quite close to what is desired, a fact that has been confirmed by the surgeons at the Hospital for Sick Children.

### 4.3 Implementation Notes

Our optimized implementation uses dictionaries to store the \( k-1 \)th cuts segment endpoints as potential segment start points for the \( k \)th cut. A vectorized search of the cost table uses these start points to return all feasible endpoints and their associated costs. Thus, only the feasible segments are tested, and the number of empty loop iterations is greatly reduced.

One of the key computation optimizations was precomputing the entire cost table using a graphics card (GPU). The cost with respect to \( c \) of each segment pairing can be calculated independently, allowing this step to be heavily threaded and GPU accelerated with OpenCL 1.2.

Even with this improvement, cost table calculation still accounts for 85% of all the computation time. These improvements result in massive speedups. The naïve implementation solves Sagittal 3 in 219.2s, while the optimized implementation solves it in 3.5s. This speedup is important as 10s is about the limit of end-user attention [13].

### 5 Hardness in Oracle Model

We will show that the (CR) problem with rearrangement is NP-hard when the deformed curve is explicitly discretized and the ideal curve is implicitly discretized. We will use a reduction from the NP-complete Partition problem [10].

---

\[ ^2 \text{Note that in the figures, the ends of the patient’s bandeau are counted as cuts, thus a figure with } k \text{ cuts has } k + 2 \text{ locations marked.} \]
In an instance of Partition the input is a multiset of \( n \) positive integers \( A = \{a_1, \ldots, a_n\} \). The goal is to decide if there exists a partition of \( A \) into multisets \( A_1, A_2 \) such that
\[
\sum_{a \in A_1} a = \sum_{b \in A_2} b = \frac{1}{2} \sum_{c \in A} c.
\]

**Theorem 2.** The (CR) problem with rearrangement is NP-hard in the model where the deformed curve is given via an explicit discretization and the ideal curve is given via an implicit discretization.

**Proof.** (Of Theorem 2) Let \( A = \{a_1, \ldots, a_n\} \) be an instance of Partition. We may assume that \( n \geq 2 \), otherwise \( A \) is trivially a No instance. Let \( A' \) be the multiset such that \( A' = \{2a_1, \ldots, 2a_n\} \). Let \( s := \sum_{a' \in A'} a' \). Note that clearly \( s \) is even. Construct an instance of (CR) as follows. Let \( f : [0, s] \rightarrow \mathbb{R} \) be the deformed curve given by the explicit discretization where its feasible cut positions are \( P = \{p_0, p_1, \ldots, p_n\} \) where \( p_0 = 0 \) and for each \( i \in [n] \) \( p_i = \sum_{j=1}^i 2a_i \). The function \( f \) is defined so that \( f(x) = 0 \) for all \( x \in [0, s] \). We can encode \( P \) in \( O(n \log \max_{i \in [n]} a_i) \) bits, and thus polynomially in the input size to the partition instance. There clearly is a polynomial time oracle for evaluating \( f(x) \).

Let \( t = s + 2 \) and let \( g : [0, t] \rightarrow \mathbb{R} \) be the ideal curve given by the implicit discretization with bound \( t \) and \( \delta = 1 \). We define \( g \) to be the piecewise linear interpolation of
\[
\{(0, 0), (1, 0), \ldots, (s/2, 0)\} \cup \{(1 + s/2, 1)\} \cup \{(2 + s/2 + 2, 0), (3 + s/2, 0), \ldots, (t, 0)\}.
\]
Clearly we can evaluate \( g \) on points in \([0, t] \cap \mathbb{Z}\) in polynomial time, and \( t \) can be encoded in a polynomial number of bit. For our cost function \( c \) we will use the same cost function presented in subsection 4.1 with parameter \( \alpha = 0 \). Our number of cuts will be \( k := n \). Choose \( \gamma = 0 \). We will denote this instance of (CR) we have constructed by \( I \). We claim that \( A' \) is a Yes instance of Partition if and only if the cost of an optimal solution to \( I \) is 0. Since \( A' \) is a Yes instance if and only if \( A \) is a Yes instance, showing this claim will suffice to complete the proof.

First suppose that \( A' \) is a Yes instance. Let \( (A'_1, A'_2) \) be a partition of \( A' \) where \( A'_1 = \{b^1_1, \ldots, b^1_n\} \) and \( A'_2 = \{b^2_1, \ldots, b^2_n\} \).

For each \( h \in [2] \), let \( \sigma_h : [n] \rightarrow [n] \) be any permutation satisfying that for all \( i \in [n] \) if \( 2a_i \in A'_h \) then \( \sigma_h(i) = j \) where \( 2a_i = b^h_j \) is in \( A'_h \).

Let \( p_i = d_i \) for each \( i \in [n] \cup \{0\} \). This is a feasible set of cut positions in \( P \). Note that \( p_i - p_{i-1} = 2a_i \) for each \( i \in [n] \). For each \( i \in [n] \) such that \( 2a_i \in A'_1 \) choose
\[
{r_i := \sum_{j=1}^{\sigma_1(i)} b^1_j}
\]
and
\[
{\ell_i := \begin{cases} \sum_{j=1}^{\sigma_1(i)-1} b^1_j, & \text{if } \sigma_1(i) > 1 \\ 0, & \text{otherwise.} \end{cases}}
\]
Similarly for each \( i \in [n] \) such that \( 2a_i \in A'_2 \) choose
\[
{r_i := 2 + \sum_{j=1}^{\sigma_2(i)} b^2_j}
\]
and
\[
{\ell_i := \begin{cases} 2 + \sum_{j=1}^{\sigma_2(i)-1} b^2_j, & \text{if } \sigma_2(i) > 1 \\ 0, & \text{otherwise.} \end{cases}}
\]
Observe that by construction, and since \( A' \) is a Yes instance, the segments of \( f \) corresponding to elements of \( A'_1 \) are mapped to \( g_{[0,s/2]} \) without violating the overlap constraint (i.e. \([\ell_i, r_i] \cap [\ell_j, r_j] \) is either a singleton or empty). Similarly the segments of \( f \) corresponding to elements of \( A'_2 \) are mapped to \( g_{[s/2+2,t]} \) without violating the overlap constraint. Thus the total cost of the solution presented is 0, and since \( c \) is non-negative this is optimal.
Now suppose that instance $I$ of (CR) has an optimal solution of cost 0. Let $p_0, p_1, p_2, \ldots, p_n \in P$ and $(\ell_1, r_1), \ldots, (\ell_n, r_n) \in Q$ define this solution. Since $k = n$, the only feasible cut positions are such that $\{p_0, p_1, p_2, \ldots, p_n\} = P$. Thus we have cut $f$ into $n$ segments, where the $i$th segment is of length $2a_i$. Since $2a_i > 1$ for all $i$, if
\[ \text{cov}(\ell_i, r_i) \cap (s/2, 2 + s/2) \neq \emptyset \]
then the cost with respect to $c$ of mapping the $i$th segment will be greater than 0, contradicting the cost of the optimal solution since $c$ is non-negative. Hence for all $i$, either
\[ 0 \leq \ell_i, r_i \leq \frac{s}{2} \quad \text{or} \quad 2 + \frac{s}{2} \leq \ell_i, r_i \leq t. \]
Let $A'_1 = \{2a_i : 0 \leq \ell_i, r_i \leq \frac{s}{2}\}$ and let $A'_2 = \{2a_i : 2 + \frac{s}{2} \leq \ell_i, r_i \leq t\}$. By the above observation, $(A'_1, A'_2)$ is a partition of $A'$. Furthermore, by construction, there is no overlap in the mapping and thus
\[ \sum_{b \in A'_1} b = \sum_{b \in A'_2} b = \frac{s}{2}. \]
Therefore $A'$ is a Yes instance as desired.

From the proof of Theorem 2 we see that this class of (CR) problems are NP-hard even when the optimal cut positions are known or fixed as problem input. It remains open to show hardness in the case of (CR) problems where both curves are explicitly discretized, with or without rearrangement.

6 Conclusion

The work described in this paper is part of a larger push to improve craniofacial surgical methods through applied mathematics and engineering. The algorithm from Section 3 is integrated into a pre-operative planning tool, allowing surgeons to pre-plan cut locations. The output of the algorithm and the planning tool is designed to interface with a projection system that shows the cut locations directly on the patient in the operating room. A prototype system is nearly complete at this stage.

In this paper we presented a formal optimization model, and algorithm for the craniofacial surgical problem of reshaping the front-orbital bar. We demonstrated its application to several test cases. Our work on the 2D setting continues, and we are currently pursuing extensive clinical testing, and in particular a postoperative, and comparative evaluation of the quality of our solutions.

Future work will focus on 3D, anterior skull craniosynostosis cases, where surgical incisions and optimization problems are much more complex.
References

1. Anantheswar, Y., Venkataramana, N.: Pediatric craniofacial surgery for craniosynostosis: Our experience and current concepts: Part-1. Journal of pediatric neurosciences 4(2), 86 (2009)
2. Burke, E.K., de Causmaecker, P., Berghe, G.V., Landeghem, H.V.: The state of the art of nurse rostering. Journal of scheduling 7, 441–499 (2004)
3. Cardoen, B., Demeulemeester, E., Béliën, J.: Operating room planning and scheduling: A literature review. European Journal of Operational Research 201(3), 921–932 (2010)
4. Cayirli, T., Veral, E.: Outpatient scheduling in health care: a review of literature. Production and operations management 12(4), 519–549 (2003)
5. Chetverikov, D., Svirko, D., Stepantsov, D., Krsek, P.: The trimmed iterative closest point algorithm. In: Proceedings of the International Conference on Pattern Recognition. pp. 545–548 (2002)
6. Cho, M., A.Kane, A., Seaward, J.R., Hallac, R.R.: Metopic “ridge” vs. “craniosynostosis”: Quantifying severity with 3d curvature analysis. Journal of Cranio-Maxillofacial Surgery 44(9), 1259–1265 (2016)
7. Frimodig, S., Schulte, C.: Models for radiation therapy patient scheduling. In: International Conference on Principles and Practice of Constraint Programming. pp. 421–437. Springer (2019)
8. Gower, J.C., Dijksterhuis, G.B.: Procrustes Problems. Oxford University Press (2004)
9. Isaac, K.V., Koenemann, J., Fukasawa, R., Qian, D., Linhares, A., Saber, N.R., Drake, J., Forrest, C.R., Phillips, J.H., Nguyen, P.D.: Optimization of cranio-orbital remodeling: Application of a mathematical model. Journal of Craniofacial Surgery 26(5) (2015)
10. Karp, R.M.: Reducibility among combinatorial problems. In: Complexity of computer computations, pp. 85–103. Springer (1972)
11. Lee, E., Yuan, F., Templeton, A., Yao, R., Kiel, K., Chu, J.C.H.: Biological planning for high-dose rate brachytherapy: Application to cervical cancer treatment. INFORMS Journal on Applied Analytics 43(5), 462–476 (2013)
12. Leitão, H.C.G., Stolfi, J.: A multiscale method for the reassembly of two-dimensional fragmented objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(9), 1239–1251 (2002)
13. Nielsen, J., Mack, R.I., et al.: Usability inspection methods, vol. 1. Wiley New York (1994)
14. Panchal, J., Utchin, V.: Management of craniosynostosis. Plastic and reconstructive surgery 111(6), 2032–48 (2003)
15. Ruiz-Correa, S., Sze, R.W., Starr, J.R., Lin, H.J., Speltz, M.L., Cunningham, M.L., Hing, A.V.: New scaphocephaly severity indices of sagittal craniosynostosis: A comparative study with cranial index quantifications. Cleft Palate-Craniofacial Journal 43(2), 211–221 (2006)
16. Saber, N.R., Phillips, J., Looi, T., Usmani, Z., B, J., Drake, J., Kim, P.C.: Generation of normative pediatric skull models for use in cranial vault remodeling procedures. Child’s Nervous System 28(3), 405–410 (2012)
17. Slater, B.J., Lenton, K.A., Kwan, M.D., Gupta, D.M., Wan, D.C., Longaker, M.T.: Cranial sutures: a brief review. Plastic and reconstructive surgery 121(4), 170e–178e (2008)
18. Wikipedia: Craniosynostosis. https://en.wikipedia.org/wiki/Craniosynostosis, online, accessed Nov 18, 2019