Constrained Path Optimisation for Underground Mine Layout

M. Brazil P.A. Grossman D.H. Lee J.H. Rubinstein D.A. Thomas
N.C. Wormald *

Abstract—**The major infrastructure component required to develop an underground mine is a decline, which is a system of tunnels used for access and haulage. In this paper we study the problem of designing a decline of minimum cost where cost is a combination of development and haulage costs over the life of the mine. A key design consideration is that the decline must be navigable to trucks and mining equipment, hence must satisfy a gradient and turning circle constraint. The decline is modelled as a mathematical network that captures the operational constraints and costs of a real mine, and is optimised using geometric techniques for constrained path optimisation. This procedure to find the optimal decline has been automated in a new version of a software tool, Decline Optimisation Tool, DOT™. A case study is described comparing this version with the earlier one.**

Keywords: optimisation, underground mining, mine layout

1 Introduction

The dominant working structure of an underground mine is a network of interconnected tunnels or mine development called drives (horizontal tunnels) and ramps (inclined tunnels) and vertical haulage shafts which provide access to ore-zones and conduits for the transport of ore to the surface. Typically, deep and long-life mines warrant shaft haulage. However truck haulage, while more expensive per unit material transported to the surface, has the advantage of earlier recovery of ore in the life of a mining project and requires only progressive capital expenditure matched to material flow.

Most open-pit mine designs are developed in an optimisation framework traceable back to the method launched by Lerchs and Grossman [3]. However, the complexity of the underground mine design problem and the unique mine design solutions sought for each ore body suggest that there will never be an elegant solution method analogous to that which exists for open-pit mining.

The major infrastructure component required to develop an underground mine is a decline (a system of ramps and drives with path topology) connecting the access and draw points with the surface portal (exit) or breakout from existing mine infrastructure. In this paper we study the problem of designing the decline so as to minimise associated costs.

More specifically, the aim is to minimise the cost of the decline for a combination of development and haulage costs corresponding to a project or life-of-mine cost. The location of the mineralised zone is determined by geological observations and surface drilling. This is followed by an infill drilling program from underground from which can be determined the location and boundaries of the orebody (i.e., that part of the mineralised zone deemed to be commercially viable), and the contained tonnages of ore. When the orebodies are located in space the set of access points (points which must be accessed for drilling and blasting operations) and draw points (from which the ore is drawn) on a sequence of levels can also be determined.

This means that in designing an optimal decline, we can assume that the access points and draw points are given. In addition, there are a number of other restrictions on the decline paths. The decline must, for example, stand off from the orebody by some specified minimum distance. Another key design consideration is that the decline must be navigable by trucks and mining equipment; this limits the gradient and curvature of the decline. We assume homogeneous ground conditions.

A procedure to find the optimal decline has been automated by the authors in a software tool, Decline Opti-

*This research has been conducted with financial support from the Australian Research Council and Newmont Australia Limited via a Linkage Collaborative Research Grant. Nicholas Wormald is also supported by the Canada Research Chairs program. We acknowledge the contributions of Andrew Fox, Nadine Wetzel and Simon Kusabs of Newmont Australia Limited for their work in clarifying Newmont's software requirements, providing valuable feedback on prototype versions of DOT2, and supplying us with real mine data to test DOT2. **Addresses:** M. Brazil, P.A. Grossman, D.A. Thomas: ARC Special Research Centre for Ultra-Broadband Information Networks (CUBIN), an affiliated program of National ICT Australia, Department of Electrical and Electronic Engineering, The University of Melbourne, Victoria 3010 Australia; D.H. Lee, J.H. Rubinstein: Department of Mathematics and Statistics, The University of Melbourne, Victoria 3010 Australia; N.C. Wormald: Department of Combinatorics and Optimization, University of Waterloo, Waterloo ON N2L 3G1, Canada. **Contact Email:** M. Brazil: brazil@unimelb.edu.au

misation Tool, DOT^{TM}. In an earlier paper [1] the first version of DOT^{TM} (which we will refer to as DOT1) is described and a case study given. In this paper we model the problem of designing an underground mine decline as a constrained path optimisation problem. We then develop a new theory of paths in 3-dimensional space that are optimal with respect to gradient and curvature constraints. This allows us to describe the principles underlying a new version of the software tool, DOT2, which replaces the heuristic methods used by DOT1 with a method based on an understanding of exact solutions to this constrained 3-dimensional path problem. DOT2 represents a significant improvement over the old version in terms of both the accuracy of the solution and the speed. A case study comparing the two versions of DOT^{TM} is described in the final section of the paper.

2 The Network Model of a Decline

The basic form of a decline, ignoring ventilation infrastructure and alternative means of egress, is a connected network forming a path from the deeper level access points to the surface portal or breakout point. Hence the decline is modelled as an underground mine network that captures the operational constraints and costs of a real mine. In the network model the nodes include the ore-zone level access points and draw points and the surface portal of the mine. The links in the model correspond to the centrelines of ramps and drives. In designing the decline, typically there are no-go regions that must be avoided such as old mine workings or ore bodies which could be sterilised by infrastructure development. In this paper we will limit the problem to designing a decline without barriers. These preliminary strategic designs allow the decline a “free” path from the breakouts to perimeter access – a best case scenario. Barrier constraints can be added later for comparison with such a best case scenario.

The ramps and drives are expensive to build with costs typically in the order of AU\$4,000 per metre (they have large cross-sections - at least 4m $\times$ 4m). A key navigability requirement is that the absolute value of the gradient of each ramp is constrained to be within a safe climbing limit for trucks, typically in the range 1:9 to 1:6.5. Hence this decline network is gradient constrained with a given maximum absolute slope. A minimum turning radius for curved ramps is a constraint imposed by the trucks and equipment to be used in the mine and is typically in the range 15-40m. The navigability constraints are significant factors in the optimal solution and to accommodate these the decline network is modelled as a gradient-constrained and curvature-constrained network.

Path declines are fundamental components of a mine network. We concentrate on the important problem of finding a least cost, navigable decline with a given path topology. In particular, this is applicable to the case where there is a single ore-zone for a proposed new underground mine or an extension to an existing mine.

Access to the ore-zone from the decline is via horizontal tunnels known as *crosscuts*. These connect the decline to the given access or draw points, which lie on a sequence of levels. Each crosscut should meet the decline at an angle of approximately 90 degrees for geomechanical stability. At each level of the ore body a set of nodes, representing a discrete choice of junctions at which the crosscut can meet the decline, is specified. We refer to this set of nodes as a *group*. Each of these nodes has an associated fixed cost that is proportional to the length of crosscut and dependent on the tonnage of ore to be hauled along the crosscut. There is a requirement that the path decline goes through one node from each group. This notion of a group of nodes adds design flexibility and optimisation opportunities.

We can now model the optimal decline design problem as that of finding a smooth path in 3-dimensional space of minimum cost satisfying the following conditions:

1. It passes through one node from each of the groups of specified nodes at each level, in a given order; furthermore, for each node the path passes through, the horizontal planar projection of the path at that node has a direction within a user-specified range of directions;
2. At each point it has gradient at most m, where m is a given constant;
3. At each point it has radius of curvature at least c when projected into the horizontal plane, where, again, c is a given constant.

The cost function for this path depends both on development and the associated haulage costs over the life of the mine. The cost of each link is the sum of the *development cost* and *haulage cost*, where the development cost is proportional to length and the haulage cost is proportional to the length times the tonnage. In particular, the cost of a link can be minimised by minimising its length, since the tonnage of ore to be hauled for each link is fixed. For a given path the total cost is the sum of the costs of all links in the path plus the sum of the costs of the selected nodes (one from each group).

The problem is solved by discretising the set of possible directions of the path at each of the intermediate nodes. This allows us to employ a bottom-up dynamic programming strategy. Suppose there are (up to) l nodes in each group. The direction vectors at a node will have a limited number of directions entering (or leaving) the incident link depending on how close the node is to the ore zone. If the maximum number of directions at any node is n,

ISBN:978-988-98671-2-6

then at each level or stage, we keep track of the (up to) nl minimum cost subpaths constructed so far for the nodes at that level. Assuming we can construct a minimum cost link with given end-point directions in constant time, the path will be constructed in approximately $O(kn^2l^2)$ time, where k is the number of levels.

This dynamic programming framework essentially reduces the problem to one of efficiently minimising the length of a single link with given positions and directions for the start and end points.

3 Link optimisation

From Section 2, the decline optimisation problem can be reduced to the problem of optimising a single link with given start and finish directions, corresponding to "approach" and "departure" directions of that link in the path. An abstract solution to the modified problem of finding minimal paths in 3-dimensional space with given start and finish directions and a given minimal turning circle (but no gradient constraint) has been described in [4]. Unfortunately, as well as violating the gradient constraint, this solution also has the undesirable property of allowing a continually varying gradient. For the underground mine design problem it is an industry requirement that the gradient on each link is both bounded and unchanging.

We approach the problem of minimising the cost of such a link by considering the projected problem in the horizontal plane. Note first that any path in the plane can be transformed to a path with given constant gradient in 3-dimensional space, by furnishing it with a suitable uniform gradient. The length of this transformed 3-dimensional path depends only on the length of the planar path and the gradient. The transformed path satisfies the gradient constraint if, and only if, the length of the path in the plane reaches a certain lower bound B dependent on the absolute vertical distance z between the end points of the link: $B = z/m$.

Hence, it is clear that the problem of finding the minimum cost link in 3-dimensional space with given approach and departure directions and gradient and turning circle constraints can be solved by finding the shortest path in the plane with these same direction and turning circle constraints and with a lower bound on the length, and then transforming this path back to the 3-dimensional setting. Dubins solved this planar problem *without* a lower bound on length in [2]. Our method for the planar problem where the lower bound is included is based on an extension of Dubins' work.

3.1 The class of admissible paths

An appropriate mathematical setting for the study of curvature constrained paths in the plane is provided by the concept of a configuration space of paths. Given two directed points p and q in $\mathbf{R}^2$, we shall call a path P from p to q in $\mathbf{R}^2$ *admissible* if:

1. P has a continuous first derivative and a piecewise continuous second derivative, i.e., P is C^1 and piecewise C^2;
2. The tangents to P at its start and end points coincide with the directions of p and q respectively;
3. The absolute curvature of P is bounded above by a specified positive constant (which we will take to be 1 by choosing a suitable scaling).

We denote by $\mathcal{A}_{pq}$ the class of all admissible paths from p to q.

It is useful to define the class $\mathcal{S}_{pq}$ consisting of all paths in $\mathcal{A}_{pq}$ that are built using components which are either arcs of unit circles or straight line segments, smoothly joined together. The class $\mathcal{S}_{pq}$ has the structure of a stratified space. By this we mean that paths that have different numbers of components and that cannot be continuously deformed into each other within $\mathcal{A}_{pq}$ belong to different regions in $\mathcal{S}_{pq}$. Each such region is a finite dimensional manifold called a *stratum* whose closure is the union of the region itself together with lower dimensional strata in $\mathcal{S}_{pq}$. This gives a very useful setting in which to describe ways of building and extending paths. In the classical paper of Dubins [2], it is proved that shortest paths in $\mathcal{A}_{pq}$ are formed from at most three different components. In Subsection 3.2, we investigate ways of extending Dubins paths using only a small number of additional components. For these reasons, typically only a small number of strata are important for our consideration.

3.2 Dubins paths and their extensions

In this subsection, we address the problem of determining an optimal curvature-constrained path between two directed points p and q in the plane, with a prescribed lower bound B on the length of the path. Specifically, we seek a minimum length admissible path, P, from p to q in the plane, such that the length of P is at least B.

The main result of Dubins [2] is as follows.

Theorem 3.1 *Given any two directed points, p and q, in the plane, there exists an admissible path of minimum length from p to q. Further, any such path must take one of the following forms:*

- *An arc with radius 1 and length less than 2π, followed by a line segment, followed by an arc with radius 1 and length less than 2π;*

- *A sequence of three arcs with radius* 1 *and with alternating senses (i.e., left-right-left or right-left-right), where the length of the middle arc is greater than* π*, and the length of each arc is less than* 2π*.*

Note that one or more of the arcs or line segments may be degenerate, in the sense that its length is zero. We shall refer to paths of the form given in Theorem 3.1 as *Dubins paths*. For any given pair of directed points, there may be up to six Dubins paths. Using L, S and R to denote respectively a left turning arc, a (straight) line segment, and a right turning arc, we can identify each Dubins path by a unique descriptor, called its *type*: LSL, LSR, RSL, RSR, LRL and RLR. For given directed points p and q, there are always Dubins paths of types LSL and RSR, and there may or may not be a Dubins path of each of the other four types, depending on the geometric relationship between p and q. Note that the Dubins paths need not all be distinct if degenerate arcs or line segments are present. Three of the Dubins path types are illustrated in Fig. 1.

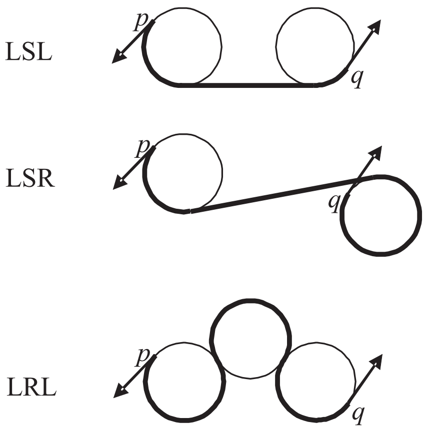

Figure 1: Three of the six types of Dubins paths; examples of the other three can be obtained from these by reflection.

Dubins paths are locally minimal: a small perturbation of a Dubins path to another admissible path cannot result in a shorter path. The converse is false; for example, a path produced by inserting a full circle into a straight section of a Dubins path is locally minimal but is not a Dubins path. Given directed points p and q, a minimum length admissible path from p to q can be found simply by calculating the lengths of each of the Dubins paths from p to q and selecting the shortest path. If the shortest Dubins path has length at least B, then it is the solution to the original problem. Suppose now that the length of the shortest Dubins path is less than B. The approach we take here is to try to obtain an admissible path with length B by extending the shortest Dubins path. We will see that this is not always possible, in which case

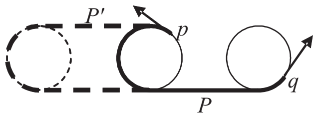

Figure 2: A parallel extension.

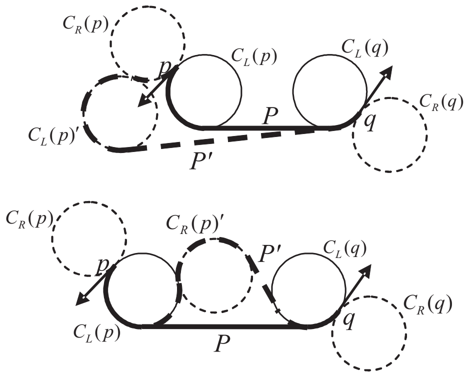

Figure 3: The two types of rolling extensions.

a solution, possibly with length greater than B, will be obtained using one of the other Dubins paths.

If a Dubins path, P, contains an arc with length at least π, then P can be extended to an admissible path P' of any greater length in the manner illustrated in Fig. 2. In particular, any Dubins path of type LRL or RLR can be extended in this way, since the length of the middle arc of any such path is greater than π. We refer to this type of extension as a *parallel extension*.

If the lengths of the arcs of a Dubins path are all less than π, then the situation is more complicated. Consider a Dubins LSL or LSR path, P, from p to q. Then P can be extended as follows. Let $C_L(p)$ and $C_R(p)$ denote the circles with unit radius that are tangent to the directed point p, on the left and right sides of p respectively. (Thus, the first arc of P is an arc of $C_L(p)$.) Similarly, let $C_L(q)$ and $C_R(q)$ denote the circles with unit radius that are tangent to the directed point q, on the left and right sides of q respectively. It is helpful to imagine P as an elastic band fixed at p and q, and the four tangent circles as barriers that restrict the region in which P can lie. Then P can be extended either by "rolling" $C_L(p)$ clockwise around $C_R(p)$, or by "rolling" $C_R(p)$ anticlockwise around $C_L(p)$, keeping the other three circles fixed, as shown in Fig. 3. We refer to these two types of extensions as *rolling extensions*.

In the configurations depicted, P could be extended indefinitely by rolling one circle sufficiently far around the other. However, as far as the present application is concerned, it makes more sense to extend P in this manner only until one of the arcs achieves a length of π, at which stage a parallel extension can be applied if necessary.

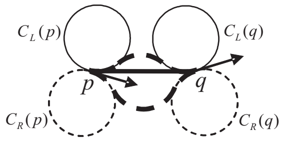

Figure 4: An example of a path that is not infinitely extendible.

For some configurations of p and q, a rolling extension can be carried out only until the extended path P' reaches a locally maximum path. If P' achieves the required length B before this occurs, then the local maximum causes no difficulties. However, if the length of the locally maximum path is less than B, then another strategy is needed. Applying the other type of rolling extension may prove successful, but this is not always the case. Fig. 4 depicts a situation where the path P' "gets stuck" between the four tangent circles. The two paths shown as broken lines are both local maxima.

We call a path *infinitely extendible* if it has arbitrarily long extensions. The configuration shown in Fig. 4 typifies the situations where there are paths that are not infinitely extendible: the distance between the centres of $C_L(p)$ and $C_L(q)$, and the distance between the centres of $C_R(p)$ and $C_R(q)$, are both less than 4. If this condition is satisfied, then there are two local maximum paths from p to q, and any admissible path that lies entirely in the region bounded by these two paths is not infinitely extendible. In particular, the region contains a unique Dubins path that is not infinitely extendible.

3.3 Local Maxima

A consequence of the previous subsection is the importance of understanding local maxima.

We first note that if P is a locally maximum admissible path in the plane between two directed points then P contains no twice differentiable points with curvature less than 1. This follows from the observation that the length of a segment of the path with curvature less than 1 can be increased (while remaining admissible) by the perturbation of a unit circle tangent to an interior point of such a segment, resulting in replacing part of the segment by three unit circle arcs.

It follows that P must be a sequence of unit circle arcs with alternating senses, which we denote by a sequence of C's. So, for example, CCC represents a path of type LRL or RLR in the notation of Subsection 3.2. The key theorem is as follows.

Theorem 3.2 *Given two directed points, p and q, in the plane, let P be an admissible path from p to q. If P is a local maximum then P is of the form CCC (or a degeneracy) where the angle around each circle is less than π and the sum of the two angles around the outer two circles minus that around the inner circle is less than π.*

The proof of the theorem involves showing that a path of the form CCCC is not a local maximum. The result is fairly straightforward, and the details are not given here.

Another important result is that the converse of Theorem 3.2 holds, giving a complete characterisation of local maxima. Again, details of this will appear in a future paper.

4 A Minimal Path algorithm

We now outline an algorithm for constructing a single minimum cost link in 3-dimensional space with given approach and departure directions and gradient and turning circle constraints, using the results of the previous sections to confirm the correctness of the algorithm. The final decline is built from these minimum cost links via dynamic programming, as discussed in Section 2, and is minimum up to the degree of discretisation of positions and angles at the nodes.

The algorithm first solves the planar path problem with lower bound B considered in Section 3. For a given link, let p and q be the projections in the horizontal plane of its two endpoints. Apart from certain degenerate configurations which we describe below, at most one of the Dubins paths from p to q can fail to be infinitely extendible. Consequently, we can solve the problem of constructing a minimal path as follows:

1. Identify the shortest Dubins path, P_1, from p to q.
2. If the length of P_1 is greater than or equal to B, return "P_1, don't extend" and stop.
3. Calculate $L_{\max}$ as follows: if there are two local maximum paths from p to q, then $L_{\max}$ is the length of the longer one, otherwise $L_{\max} = \infty$.
4. If $L_{\max} > B$, return "P_1, extend to length B" and stop.
5. Identify the second shortest Dubins path, P_2, from p to q.
6. If the length of P_2 is greater than or equal to B, return "P_2, don't extend" and stop.
7. Return "P_2, extend to length B" and stop.

If two or more Dubins paths coincide because of the presence of degenerate arcs or line segments, then the shortest and second shortest Dubins paths may be the same. If this occurs, then P_2 is the second shortest distinct Dubins path in Step 5. This strategy may fail if there is a Dubins path from p to q that consists of just a line segment, in which case the algorithm must be modified by taking P_2 to be the path obtained by appending a circle, i.e., an arc of length 2π, to P_1. Note that the algorithm assumes that if P_1 is not infinitely extendible then P_2 is infinitely extendible. A proof of this will appear in a future paper.

Observe that no paths are actually constructed in the course of running the algorithm; rather, the algorithm returns a Dubins path type together with information on whether and how the Dubins path is to be extended. Once the length L_{pq} of the optimal planar path between p and q has been computed, then it is easily seen that the length of the corresponding path in 3-dimensional space (furnished with a constant gradient) is $\sqrt{L_{pq}^2 + (z_p - z_q)^2}$, where z_p and z_q are the z coordinates of the two endpoints of the link. This strategy of computing lengths and recording the type of Dubins path, rather than constructing each path during the dynamic programming, ensures that ultimately only the links that are actually needed for the optimal decline path are constructed.

Finally, we comment briefly on the problem of constructing the links in the optimal decline path. A link is represented by a list of line segments and arcs, where each line segment is parameterized by its start and end points, and each arc is parameterized by its centre, start angle and turn angle. Constructing a Dubins path and a parallel extension are straightforward, but constructing a rolling extension is more difficult, and generally requires the use of an iterative procedure. The details are not given here.

Once a planar path has been constructed for a given link it is converted to a path in 3-dimensional space by giving it the correct constant gradient: $\pm(z_p - z_q)/L_{pq}$.

5 Case Study

DOT^{TM} has been tested mainly in design tasks for various Australian and New Zealand mines operated by Newmont Australia Limited, our collaborative research partner. This has been particularly valuable in refining the features in DOT^{TM} to match both operational and strategic design needs; DOT2 incorporates many refinements over DOT1 - mainly, the algorithm outlined in Section 4. In June 2006 we were offered the chance to compare our design with one developed by an experienced mine consultant - though, for reasons of commercial confidentiality, we cannot disclose the name of the mine. The design was required to span 18 given access or draw points with a decline maximum gradient of 1:7 and a minimum turning radius of 25 metres. We were able to compare our DOT1 and DOT2 designs against the engineer's design. DOT1 took about 20 minutes at a reasonable level of accuracy to find an unconstrained decline of 1883 metres in length. DOT2, on the other hand, took a few seconds

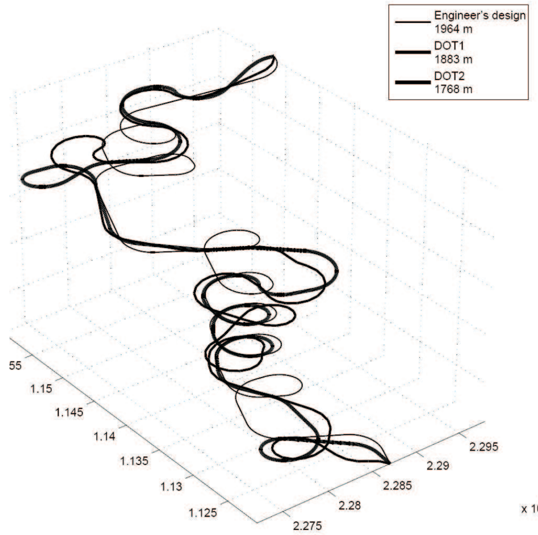

Figure 5: A comparison of the engineer's original design with the declines generated by DOT1 and DOT2.

only to find a design of length 1768 metres. The engineer's original design was 1964 metres in length. Thus DOT2 significantly outperformed DOT1 in time and total decline length but both were superior to the original design. DOT2 saved about 10% over the original design. Fig. 5 compares DOT1, DOT2 and the original design in a composite representation. Since a metre of decline development currently costs about AU$4,000, the development savings alone are of the order of AU$784,000. Over the life of a mine the corresponding savings in haulage, ventilation and other operational costs are approximately double this, leading to an overall saving of about AU$2.3 million.

References

[1] Brazil, M., Lee, D.H., Leuven, M. Van, Rubinstein, J.H., Thomas, D.A., Wormald, N.C., "Optimising Declines in Underground Mines", *Mining Technology: Trans. of the Institution of Mining and Metallurgy, Section A*, V112, pp. 164-170, 2003

[2] Dubins, L.E., "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *Amer. J. Math.*, V79, N3, pp. 497-516, 7/57

[3] Lerchs, H., Grossman, I.F., "Optimum Design of Open-Pit Mines," *Trans. Canadian Institute of Mining and Metallurgy*, V68, pp. 17-24, 1965

[4] Sussman, H.J., "Shortest 3-dimensional paths with prescribed curvature bound," *Proc. 34th Conf. on Decision and Control, IEEE*, pp. 3306-3312 12/95