University of Ljubljana
Institute of Mathematics, Physics and Mechanics
Department of Mathematics
Jadranska 19, 1111 Ljubljana, Slovenia

Preprint series, Vol. 42 (2004), 917

MINIMUM CURVATURE VARIATION B-SPLINES VALIDATION OF A PATH-PLANNING MODEL

Tomas Berglund Andrej Brodnik
Håkan Jonsson Kent Mrozek
Mats Staffanson Inge Söderkvist

ISSN 1318-4865

March 23, 2004

Ljubljana, March 23, 2004

Minimum Curvature Variation B-splines Validation of a Path-Planning Model

Tomas Berglund* Andrej Brodnik* † Håkan Jonsson*
Kent Mrozek‡ Mats Staffanson* Inge Söderkvist§

March 23, 2004

Abstract

We study the problem of computing smooth planar paths in the presence of obstacles. We investigate the relationship between smoothness of a path and its traversal time in the case of a four-wheel four-gear articulated vehicle. Smoothness is measured by minimum curvature variation, i.e., by the integral over the square of the derivative of curvature along a path. Paths are defined by B-splines or piecewise polynomials and obstacles are represented by polygonal chains. Obstacle-avoidance is achieved by means of the envelope of B-splines. We present eight real-world cases based on application data from the Swedish mining company LKAB. The cases show that our minimum curvature variation B-spline path-planning algorithm yield paths that are substantially better than the one used by LKAB today.

1 Introduction

Path-planning is a problem area that is very important as the level of automation increases [12, 14, 18, 24]. Today, there are many industrial applications that require the pre-computation of paths for autonomous vehicles and robot arms to follow and move along. In this paper, we address the problem of computing an obstacle-avoiding planar path that combines the properties of being both smooth and and fast to traverse.

The work space of autonomous vehicles and robot arms are often cluttered with *obstacles* that these machines must avoid. An example is the path-planning of the autonomous vehicles used for transporting iron ore in the underground mines of the Swedish mining company Luossavaara-Kiirunavaara

*Luleå University of Technology, Department of Computer Science and Electrical Engineering, Sweden
†IMFM, Ljubljana and University of Primorska, Faculty of Education, Slovenia
‡QNavigator AB, Luleå, Sweden
§Luleå University of Technology, Department of Mathematics, Sweden

Aktiebolag (LKAB) [19] . It is crucial that these autonomous vehicles avoid the obstacles constituted by the mine walls in order for the production to go on. Another example is a robot arm holding a cutting tool that follows a pre-computed path. It should perform its task while staying away from other machines and it should not damage itself.

Much work has been devoted to planning paths in the presence of obstacles and there are many results yielding piecewise linear paths [12, 13]. However, practical applications involving physical machines also require that the paths are *smooth* [4, 9, 16]. An example of the importance of smoothness comes from the offset path of a cutting tool [20]. Its path must be curvature-continuous to avoid jumps in the acceleration that damage the driver motors. Another example comes from considering the autonomous ore transport vehicle of LKAB. It is an articulated vehicle, which means that the whole vehicle is involved in the steering. Its steering gear is worn out even by small jerks in its path and large jerks increases the risque of slippage. Moreover, such jerks easily cause the machine to drop ore on the road which eventually must be cleared before the transports may continue.

There are many definitions of smoothness. Depending on context, a path or curve in 2D as well as in 3D is referred to as being smooth if it is tangent continuous, has a continuous curvature, or even has a continuous derivative of curvature [4, 9, 16]. Smooth curves are of importance not only to achieve paths that do not cause unnecessary wear. They have also found application in designing curves that look aesthetically pleasing. Such applications are found in the car, airplane, virtual reality, video game, and film industry [11, 21, 22]

Here, an important aspect of the smoothness of a path is its relation to the speed at which the path can be traversed. The turning speed of the transport vehicle is limited by the speed of the vehicle. The speed of the vehicle is therefore adjusted such that the vehicle is capable of following the path. If the vehicle has different gears with certain limitations on curvature and derivative of curvature, it is seen that the traversal time is a function of the gears along the path. Then, in turn, the gears are functions of the curvature and the derivative of curvature along the path. Therefore, the higher the smoothness, the higher the gear, and the faster the traversal of the path.

Still, obstacle-avoidance and smoothness is not enough for practical purposes. The path should also stay at a sufficient distance – the *safety margin* – from the obstacles. The smoother the path, the larger the radius of the curve. This means that a path will tend to go close to obstacles in order to have as high smoothness as possible. Once again, consider the ore transportation vehicle. A smooth path makes it go close to the walls of the mine when turning. As touching the walls is the main danger for the vehicle, it should stay at a safety distance from the walls. The computation of safety margins has been considered in the literature [6].

1.1 Contribution

In this paper we focus on validating the assumption that geometrically smooth paths are fast to traverse. The Swedish mining company LKAB has already planned piecewise seventh-degree polynomial paths inside their mine. Based on mine map application data from LKAB, in the form of polygonal chains, we compute safety margins that, in turn, are two flanking polygonal chains. Inside of these margins, using a standard nonlinear programming tool, we compute geometrically smooth paths built from obstacle-avoiding quartic minimum curvature variation B-spline curves. Using the same model of a four-gear four-wheel articulated vehicle with gear-limits as the one used by LKAB, we validate an increase in speed along the B-spline paths compared to the speed along the LKAB paths. This is done for eight scenarios. Output from our computations are piece-wise polynomial paths with a dedicated gear for each piece, traversal time, curvature, and derivative of curvature.

1.2 Paper outline

In the next section, we present some preliminaries. It is then followed by Section 3 which introduces the vehicle model used. Our method to produce a safety margin follows in Section 4. Section 5 shows a comparison between our paths and the LKAB paths, while Section 6 concludes this paper.

2 Preliminaries

This section contains a description of the LKAB paths and their representation, our smoothness measure, and a motivation to the use of B-splines as a path representation. It also contains a description of the minimum curvature variation B-spline (MVB) and how it is computed

Our interest lies in the properties of path cycles. A path cycle is built from paths which in turn are built from consecutive path segments. All paths must meet up with initial and terminal endpoint constraints. These are given as position, slope angle, curvature, and derivative of curvature. Furthermore, the paths must not intersect obstacles represented by flanking polygonal chains. These obstacles are given as input for the path-planning.

LKAB describes path segments by seventh-degree polynomials. These paths are not computed to minimize curvature variation over the whole path. Instead, for each polynomial segment, the maximal derivative of curvature is minimized. Each segment is defined in a separate coordinate system, which gives a path built from consecutive parameterizations along different room axes. This representation is flexible, but the constructed curves have the drawback that there is nothing that guarantees smoothness nor a small traversal time for the whole path. We compute paths that are quartic MVB curves, which are described in Section 2.3.

2.1 Smoothness measure

We consider the same smoothness measure as Kanayama and Hartman [17], namely the integral over the square of arc-length derivative of curvature along the path. A motivation of using this measure comes from the physics of motion. At constant speed, the centripetal (lateral) acceleration

$$\frac{mv^2}{r} = mv^2K$$

of a vehicle is proportional to its curvature K, where m is the mass of the vehicle, r is the radius curvature, and v is its velocity. The time derivative of this acceleration is given by

$$\frac{d}{dt}(mv^2K) = mv^2\frac{dK}{dt} = mv^2\frac{dK}{ds}\frac{ds}{dt} = mv^3\frac{dK}{ds}.$$

Thus, the variation of curvature is proportional to the jerk or change in lateral acceleration of the vehicle.

We are dealing with curves that are functions over a room axis. For a function $f(x)$ with curvature $K(x)$, $x \in [x_0, x_1]$, there is the one-to-one relation $ds = \sqrt{1 + f'(x)^2}\, dx$, between x and arc-length s. Therefore, letting $\dot{\nu} = d\nu/ds$ and $\nu' = d\nu/dx$, our smoothness measure can be written as

$$\int_{s(x_0)}^{s(x_1)} \dot{K}(s)^2\, ds = \int_{x_0}^{x_1} \frac{K'(x)^2}{\sqrt{1 + f'(x)^2}}\, dx.$$

2.2 B-splines for path-planning

B-spline functions [5, 7, 8] have properties that make them suitable for smooth and obstacle-avoiding path-planning. A B-spline function or B-spline $B(x)$ is a piecewise polynomial of a certain degree. Here, we consider quartic, i.e., degree 4, uniform B-splines that are functions of x. Quartic B-splines are sufficiently smooth for our purposes as they are continuous even in their derivative of curvature.

By means of its envelope [20, 23], a B-spline function can avoid obstacles represented by piecewise linear functions $\underline{c}(x) \leq \overline{c}(x)$. The envelope of $B(x)$ is a pair of piecewise linear functions $\underline{e}(x)$ and $\overline{e}(x)$, given by $B(x)$, such that $\underline{e}(x) \leq B(x) \leq \overline{e}(x)$. In order for the B-spline to avoid the obstacles it suffices to impose the constraints $\underline{c}(x) \leq \underline{e}(x)$ and $\overline{e}(x) \leq \overline{c}(x)$. This, in turn, is the key that opens the door to obstacle-avoidance of the B-spline $B(x)$. The constraints $\underline{c}(x) \leq \underline{e}(x)$ and $\overline{e}(x) \leq \overline{c}(x)$ can be formulated as a finite number of linear constraints as they amount to comparing the piecewise linear functions, representing the obstacles, at their respective vertices. Then, instead of having to compare the function itself, at an infinite number of points, against the obstacles, we have finitely many constraints depending only on the number of vertices of the envelope and the vertices of the piecewise linear obstacles.

B-splines are written in closed form, and they are also efficiently computed [7]. Another thing that makes B-splines attractive is the ease by which the shape of the resulting curve can be controlled. Their approximation ability and the properties of their shape depend on the number of B-spline basis functions used to define $B(x)$ [3]. The larger number of basis functions $B(x)$ is built from, the more flexible its representation. For these reasons, B-splines are also widely used in a variety of contexts such as data fitting, computer aided design (CAD), automated manufacturing (CAM), and computer graphics [11].

2.3 Minimum curvature variation B-spline paths

We are concerned with obstacle-avoiding minimum curvature variation B-splines (MVB) [1, 2]. They are solutions to the following problem:

Problem 1 *Given as input two flanking obstacles; polygonal chains that are piecewise linear functions $\underline{c}(x) \leq \overline{c}(x)$, defined on a real interval $I = [x_0, x_1]$, certain endpoint constraints, and a certain number, n, of B-spline basis functions. Compute a quartic uniform B-spline $B(x)$ defined on I such that*

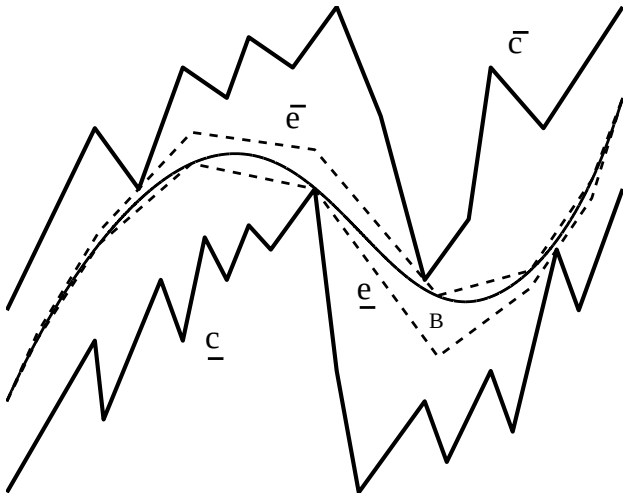

Figure 1: A minimum curvature variation B-spline (MVB) B with its envelope $\underline{e} \leq \overline{e}$ between two polygonal chains $\underline{c} \leq \overline{c}$ such that $\underline{c} \leq \underline{e} \leq B \leq \overline{e} \leq \overline{c}$.

a) The B-spline $B(x)$ satisfies the given endpoint constraints.
b) $\underline{c}(x) \leq \underline{e}(x)$ and $\overline{e}(x) \leq \overline{c}(x)$.
c) The B-spline $B(x)$ minimizes the cost function

$$\int_I \frac{K'(x)^2}{\sqrt{1 + B'(x)^2}}\, dx$$

where $K(x) = B''(x)/(1 + B'(x)^2)^{\frac{3}{2}}$ is the curvature and $\underline{e}(x) \leq \overline{e}(x)$ is the envelope of $B(x)$ respectively.

This problem is an n-dimensional nonlinear optimization problem with linear constraints that can be solved with standard optimization tools. Figure 1 shows an MVB that is a solution to a setting where the B-spline is built from 10 basis functions. The drawback of this problem formulation is that it only accounts for obstacles that are linear functions on a room axis. Therefore, when a setting can not be parameterized along a room axis, this problem can be split into subproblems giving a suboptimal solution as a combination of the solutions to the subproblems.

3 Vehicle model

We use a vehicle model based on the Load-Haulage-Dump vehicle Tamrock Toro 2500 [25]. It is an electrical wheel loader with articulated steering. A picture of the vehicle is seen in Figure 2. Even though it is a rough approximation, we will consider the sweep area of the vehicle being of circular shape.

Figure 2: Side view of the Tamrock Toro 2500 electrical wheel loader; length: 14 meters, operating weight: 76 tonnes, and capacity: 25 tonnes.

3.1 Gears, speed, acceleration, and time

The vehicle has four gears. The power output to the wheels of the vehicle is about 165kW on average. The vehicle weighs differently depending on the ore load in its bucket. We consider an average weight of 85 tonnes.

A mean driving speed of each gear and the mean acceleration up to each gear has been measured at LKAB according to Table 1. The deceleration is 0.9 meters per second for all gears.

Gear	**Measured speed (m/s)**	**Acceleration (m/s^2)**
1	1.0	3.8
2	1.9	1.3
3	3.1	0.8
4	5.0	0.5

Table 1: Gear speeds and accelerations.

The time it takes for a vehicle to traverse a path cycle is a sum of the time it takes for the vehicle to traverse each path segment. Each segment has a prescribed gear chosen such that it is the highest allowed gear on the segment. The speed over a segment can contain an acceleration part, a constant speed stretch and a deceleration. It is important that accelerations are not made before entering a segment with higher gear and that decelerations are made before entering a segment with lower prescribed gear.

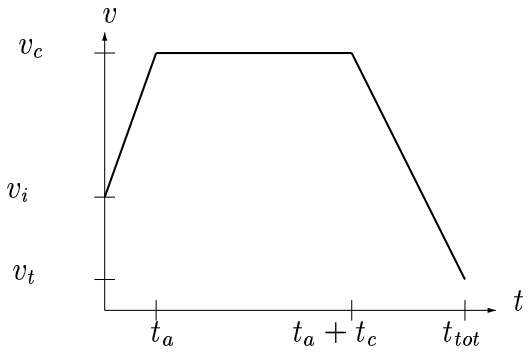

Figure 3: Time calculation of a path.

Let a be the acceleration, d the deceleration, and v_c the constant speed. Furthermore, let v_i, v_t, and l be the initial speed, terminal speed, and the length of the segment, respectively, see Figure 3. Then, the time $t_a = (v_c - v_i)/a$ of acceleration and the time $t_d = (v_c - v_t)/d$ of deceleration can be used to compute the arc-lengths s_a, s_d, and s_c of acceleration, deceleration, and constant speed, respectively, through

$$s_a = \frac{at_a^2}{2} + v_i t_a, \quad s_d = \frac{dt_d^2}{2} + v_t t_d, \quad s_c = l - s_a - s_d.$$

The total time t_{tot} is then, after some computation, shown to be

$$t_{tot} = t_a + \frac{s_c}{v_c} + t_d = \frac{(v_c - v_i)^2}{2av_c} + \frac{l}{v_c} + \frac{(v_c - v_t)^2}{2dv_c}$$

The time t_c and the length s_c are allowed to be negative. This occurs if the constant speed is never reached on the segment.

3.2 Limits in maneuverability

The vehicle has limits in its maneuverability. When designing paths, the steering angle α of the vehicle is limited to $|\alpha| \leq \alpha_{\max} = 38°$and the time derivative $d\alpha/dt$ of the steering angle is limited to $|d\alpha/dt| \leq \dot{\alpha}_{\max} = 10°$ per second.

A model of the vehicle is built from its technical specification, see Figure 4. The distance $L = 2.55$ meters to the joint is the same for both the front and

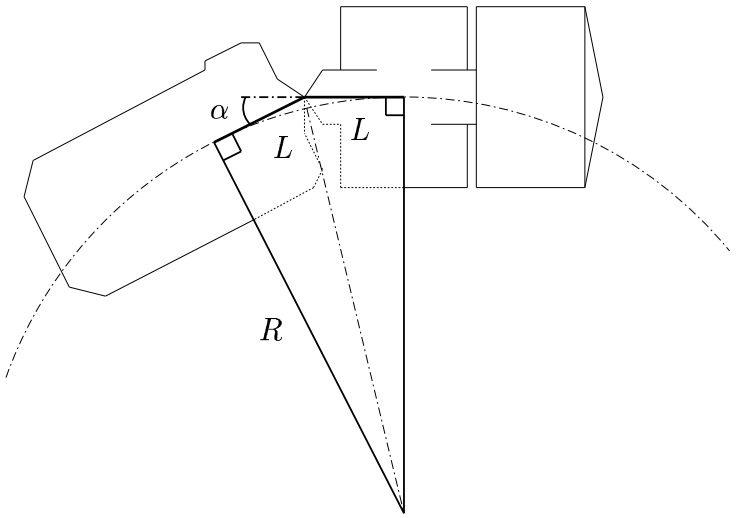

Figure 4: Model of the wheel loader.

the rear wheels. The radius of curvature is denoted R. The curvature K can be calculated as

$$K = \frac{1}{R} = \frac{\tan(\alpha/2)}{L}. \tag{1}$$

Then each path segment must fulfill

$$\left|\tan^{-1}(LK)\right| \leq \frac{\alpha_{\max}}{2}.$$

Let v be the velocity of the vehicle. From (1) the derivative of curvature is given by

$$\frac{dK}{ds} = \frac{(1+\tan^2(\alpha/2))}{2L}\frac{d\alpha}{ds} = \frac{(1+L^2K^2)}{2L}\frac{d\alpha}{dt}\frac{dt}{ds} = \frac{(1+L^2K^2)}{2Lv}\frac{d\alpha}{dt}$$

Then it is possible to isolate $d\alpha/dt$ and establish the bound

$$\left|\frac{v}{(1+L^2K^2)}\frac{dK}{ds}\right| \leq \frac{\dot{\alpha}_{\max}}{2L}$$

For each path segment, the gear with the highest prescribed velocity v is chosen such that this bound still holds.

4 Safety margin

In order to account for the sweep area of the vehicle and at the same time impose a safety distance between the vehicle and the mine walls, we compute a safety margin. Therefore, when we refer to a safety margin, we mean a margin that takes into account both the sweep area of the vehicle and a safety distance. This section contains a description of how the margin is computed.

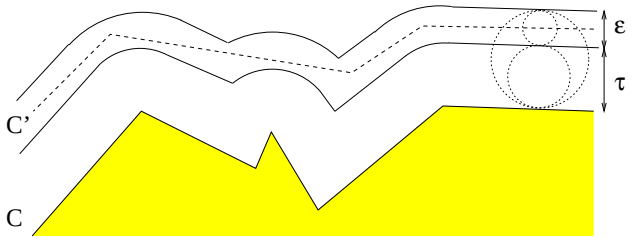

Figure 5: The polygonal chain C' approximates C. The minimal distance between a point on C' and a point on C lies between τ and $\tau + \epsilon$.

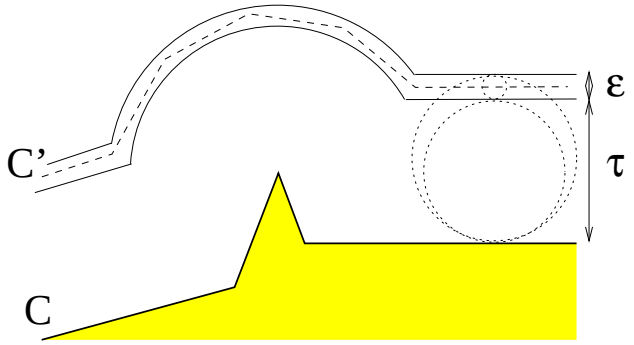

Figure 6: The polygon approximation algorithm does not guarantee a reduction in the number of vertices of the resulting polygonal chain (dashed) when ϵ is small compared to τ.

4.1 Safety margin problem formulation

Since we consider the vehicle being of circular shape, we use the Minkowski sum [6] on a polygonal chain to account for the size of a vehicle and to impose a safety margin. The result, in turn, needs to be a polygonal chain for use in our safety margin optimization problem. In order to decrease the time complexity of the path-planning, the number of vertices in the new polygonal chain should be minimized. Our problem can be formulated as:

Problem 2 *Given a polygonal chain C construct another polygonal chain C', containing no point closer to C than τ and no point farther from C than $\tau + \epsilon$, having as few vertices as possible.*

A sketch of the problem is shown in Figure 5. We solve a restricted version of this problem where the resulting polygonal chain has its vertices at distance $\tau + \epsilon/2$ from C. Our resulting polygonal chain is the solution to a polygon approximation problem on β, that is a polygonal chain built from sufficiently frequent sample points at distance $\tau + \epsilon/2$ from C.

4.2 Polygon approximation

Let $d(p, p')$ be the Euclidean distance between the points p and p'. Furthermore, let the measure $D(\cdot, \cdot)$ between the two polygonal chains V and V' be defined by $D(V, V') = \max_{p \in V} \min_{p' \in V'} d(p, p')$. The polygon approximation problem is:

Problem 3 *Given a polygonal chain V with n vertices and an error bound δ, find a polygonal chain V', consisting of a minimal length subsequence of the vertices of V, such that $D(V, V') \leq \delta$.*

A solution to this problem is presented by Iri and Imai [15]. They build a graph G by extending V with edges for all valid shortcuts from one vertex $v_i \in V$ to another vertex $v_j \in V$. A shortcut is said to be valid if all vertices $v_k \in V$, $k = i, \ldots, j$, are at distance less than δ from the straight line connecting v_i and v_j. Finding V' with minimum number of vertices is equivalent to finding the minimum number of edges in G that connect v_1 and v_n. Since G is a directed and acyclic graph, this can be done in a straightforward manner using, for example, dynamic programming techniques.

We apply the polygon approximation solution proposed by Iri and Imai [15] to our restricted problem by letting their original polygonal chain V equal our sampled polygonal chain β and their error bound δ equal $\epsilon/2$, see Figure 5. Note that there is no guarantee for our resulting polygonal chain to have less vertices than the original chain. A special case where a high number of vertices is needed is seen in Figure 6.

5 Path cycle comparisons

Our algorithm permits us to take two flanking polygonal chains and produce a safety margin. From this margin, we are able to compute, by means of its envelope, an obstacle-avoiding MVB path. This is also done in previous work [1]. Now, we want to compare our paths with corresponding paths produced in industry. We use the same vehicle and time model, see Section 3, as LKAB and compare the traversal time of MVB paths with the traversal time of the LKAB paths. The comparison is made without taking into account any control algorithm.

The LKAB paths are built from consecutive path segments. Each segment is a seventh degree polynomial over a separate room axis. This means that the LKAB path is flexible and need not be entirely defined over a certain room axis. The MVB path needs to be defined over one room axis only. The path becomes a fourth degree piecewise polynomial over the particular axis. Each piece performs a path segment. The idea is to have the same parametrization from initial to terminal endpoint. If this is not possible, the path is split into subpaths where the optimization is performed on each subpath. This yields a suboptimal path. The LKAB path representation is

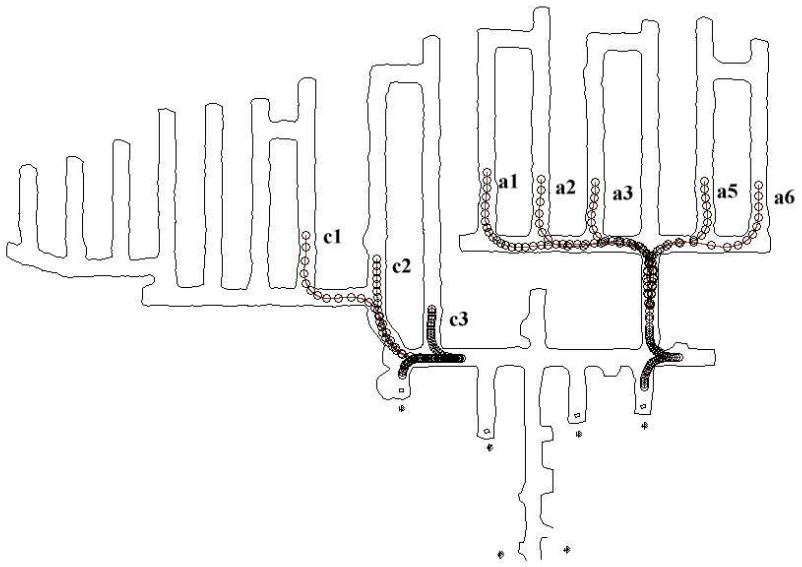

Figure 7: Mining area 820:39 and cycles a1, a2, a3, a5, a6, c1, c2, and c3.

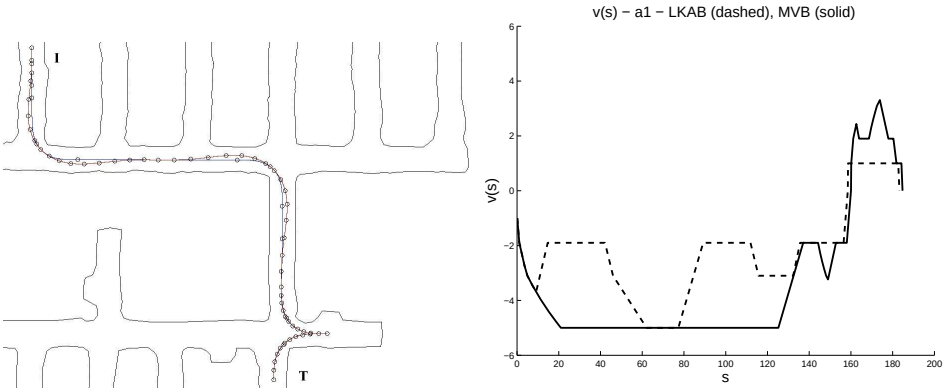

Figure 8: Cycle a1; left: LKAB and MVB cycle; right: velocity over curve length comparison scheme; LKAB (dotted) and MVB (solid).

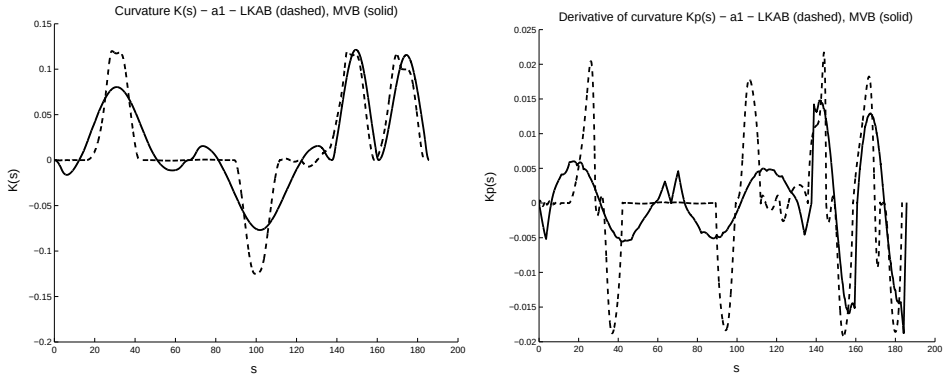

Figure 9: Cycle a1; left: curvature plot; right: derivative of curvature plot; LKAB (dotted) and MVB (solid).

more flexible than the MVB path representation. It allows the path to turn in any direction and each segment has more degrees of freedom than that of an MVB path segment. The good thing with the MVB path is that it is defined in such a manner that it yields a smooth path from initial to terminal endpoint.

We consider mining area 820:39 in the LKAB underground mine in Kiruna, Sweden. This mining area is shown in Figure 7. The figure also points out eight different path cycles, namely a1, a2, a3, a5, a6, c1, c2, and c3. A safety margin is computed such that it accounts for both the sweep area of the vehicle and a safety distance. The margin is such that it is not closer to the walls than the minimum distance from the walls to the LKAB paths. In our setting, the safety margin lies in a region between 2.25m and 2.35m from the polygonal chains describing the walls of the mine.

The MVB paths are constrained to have the same position, slope angle, curvature, and derivative of curvature at its endpoints as the LKAB paths. The derivative of curvature and the curvature are computed numerically at 300 positions along each segment. Based on this, the highest gears possible are chosen on each segment according to methods described in Section 3.2. The same method of choosing gears is used both for the LKAB and the MVB paths. We accelerate and decelerate in the same manner as LKAB and we use an average weight – 85 tonnes – in all our computations. Acceleration and a deceleration is modelled according to Section 3.1.

In order to produce an MVB path cycle, the cycle is divided into paths that are produced from 25 B-spline basis functions. For each path, knots are placed uniformly. For the optimization, the MATLAB solver `fmincon` is used with termination criteria `tolfun=tolcon=tolx=` 10^{-8}. The computation of function and gradient values depend on the integration routine `coteglob` with accuracy 10^{-8} [10]. The appropriate gears of the vehicle are chosen for each segment after the optimization.

Due to shortage of space, we present only one path cycle comparison in detail, namely the one for cycle a1, see Figure 7. Figure 8 shows path cycle a1. It also shows how velocities (gears) are chosen along the LKAB and the MVB path. Computations starts from initial position I to terminal position T. It is seen that the MVB path cycle allows a faster traversal than the LKAB path cycle. Figure 9 shows a comparison between the curvature and a comparison between the derivative of curvature for the LKAB and the MVB paths over path cycle a1. It is seen that neither the magnitude of curvature nor the magnitude of derivative of curvature of the MVB path is larger than for the LKAB path.

Cycle		$Time$	S_{cost}	$K_{\max}$	$K_{\max}$	Length
a1	LKAB	90.8	0.01358	0.125	0.02171	183.2
	MVB	56.63	0.007523	0.1214	0.01876	184.7
	Diff (%)	**37.63**	**44.62**	**2.914**	**13.58**	**-0.8301**
a2	LKAB	84.95	0.01369	0.125	0.02171	155.9
	MVB	52.89	0.007727	0.1214	0.01876	157.3
	Diff (%)	**37.74**	**43.58**	**2.914**	**13.58**	**-0.8684**
a3	LKAB	76.95	0.01374	0.125	0.02171	130
	MVB	58.11	0.01033	0.1214	0.01885	131.6
	Diff (%)	**24.48**	**24.82**	**2.912**	**13.16**	**-1.245**
a5	LKAB	75.7	0.01373	0.131	0.02171	129.6
	MVB	54.08	0.00968	0.1214	0.01876	130.3
	Diff (%)	**28.56**	**29.49**	**7.365**	**13.58**	**-0.5315**
a6	LKAB	82.52	0.01178	0.12	0.02171	151.8
	MVB	50.11	0.007129	0.1214	0.01876	152.1
	Diff (%)	**39.28**	**39.48**	**-1.136**	**13.58**	**-0.2491**
c1	LKAB	60.41	0.01077	0.1278	0.02194	112.1
	MVB	44.06	0.005326	0.1142	0.01787	112.8
	Diff (%)	**27.08**	**50.55**	**10.6**	**18.54**	**-0.6826**
c2	LKAB	40.3	0.004429	0.1278	0.02194	79.7
	MVB	27.2	0.001884	0.1142	0.01787	80.75
	Diff (%)	**32.51**	**57.45**	**10.6**	**18.54**	**-1.32**
c3	LKAB	42.58	0.006361	0.1278	0.02194	63.56
	MVB	29.9	0.004048	0.1142	0.02188	63.42
	Diff (%)	**29.78**	**36.36**	**10.6**	**0.2729**	**0.2281**
Mean	**Diff (%)**	**32.13**	**40.79**	**5.847**	**13.10**	**-0.6873**

Table 2: Measure comparisons for cycles a1, a2, a3, a5, a6, c1, c2, and c3.

We get similar results for the other seven path cycles. We present corresponding results in Table 2. The columns of the table shows the following properties:

Time The time in seconds it takes for the vehicle to traverse a path from an initial to a terminal endpoint.

Smoothness cost $S_{\mathbf{cost}}$ The smoothness cost $\int \dot{K}(s)^2 ds$ over the path. This is the measure of our optimization.

Maximal curvature $K_{\max}$ The maximal magnitude of the curvature along the path.

Maximal derivative of curvature $\dot{K}_{\max}$ The maximal magnitude of the derivative of curvature along the path.

Length The length in meters of the path.

The rows of the Table 2 are divided into groups depending on what cycle they belong to. Each cycle has a row for LKAB, MVB, and a difference in percents going from the LKAB value to the MVB value. The very last row contains a mean of all the percentage differences. It is seen that an MVB path is more smooth and faster to traverse than the LKAB path at the expense of being slightly longer. The computation time for an optimization increases with the number of B-spline basis functions. An optimization with 25 basis functions takes about 30 minutes on a Pentium IV 2 GHz machine with 2 Gb RAM memory. As seen in Figure 7, the cycles presented here mostly consists of turns. Longer cycles also include straight stretches. For these stretches, the benefit of using MVB paths is not significant as they can be traversed on the highest gear. Then, the time differences remain the same, but the relative differences decrease. It is in the turns where the MVB paths come to their best.

6 Conclusions and future work

Using application data from the Swedish mining company LKAB, we have shown that quartic minimum curvature variation B-splines can be used as a path representation. We have also shown, for eight path cycles, that these smooth B-spline curves yield paths that are faster to traverse than paths used by LKAB today. We conclude that, the higher the smoothness, the higher the gear and therefore, the shorter the traversal time.There seems to be a trade-off between short traversal times and high wear of the vehicle. A smooth path allows low wear, but it also allows a higher speed of the vehicle, which in turn increases the wear. This relation would be of interest to investigate.

Short path segments allows high granularity in the choice of gear. Low gears need not be used for a longer time than necessary. This also contributes to the lowering of traversal time and is a great time benefit of the B-spline paths. In real life, frequent changes of gears yield high wear on the gearbox of the vehicle. Future work should take short segments and short gear changes

into account. It is possible to use the method of computing a safety margin according to the one described in this paper. Still, the safety margin does not take the real sweep area of the vehicle into account. This is a difficult task, but would give a more realistic setting.

The computation of B-spline paths demands the definition of initial and terminal constraints. It also demands a division of the paths in parts that are functions along a room axis. The choice of initial and terminal constraints and the choice of cutting points can be made more automatic than they are today. The placing of knot points defining the B-splines could also be made more automatic. Through a good choice of knot placement, the B-spline paths would have sufficient degrees of freedom where it is needed, while at the same time being fast to compute. In the scenarios of this paper, we are able to plan planar paths that are up to 30 percents faster to drive along than along corresponding LKAB paths. This gives us reason to believe that a minimum curvature variation B-spline constitutes a suitable path representation if traversal time is considered. It would be of interest to investigate whether minimum curvature variation B-splines could also be used in smooth obstacle-avoiding path-planning in three dimensions.

Acknowledgements

This work was supported in part by the Swedish mining company LKAB and the Research Council of Norrbotten (Norrbottens Forskningsråd) under contract NoFo 03-006.

References

[1] T. Berglund, U. Erikson, H. Jonsson, K. Mrozek, and I. Söderkvist. Automatic generation of smooth paths bounded by polygonal chains. In M. Mohammadian, editor, *Int. Conf. on Computational Intelligence for Modelling Control and Automation (CIMCA'2001)*, pages 528–535, Las Vegas, USA, July 2001.

[2] T. Berglund, H. Jonsson, and I. Söderkvist. An obstacle-avoiding minimum variation b-spline problem. In *Int. Conf. on Geometric Modeling and Graphics (GMAG03)*, London, England, July 2003.

[3] T. Berglund, T. Strömberg, H. Jonsson, and I. Söderkvist. Epi-convergence of minimum curvature variation b-splines. Technical Report 2003:14, ISSN 1402-1536, Department of Computer Science and Electrical Engineering, Luleå University of Technology, Sweden, 2003.

[4] J.-D. Boissonnat, A. Cérézo, and J. Leblond. A note on shortest paths in the plane subject to a constraint on the derivative of the curvature. Rapport de recherche 2160, INRIA, 1994.

[5] M.G. Cox. The numerical evaluation of B-splines. *J. Institute of Mathematics and its Applications*, 10:134–149, 1972.

[6] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 1997.

[7] C. de Boor. *A practical guide to splines*. Springer-Verlag, 1978.

[8] P. Dierckx. *Curve and Surface Fitting with Splines*. Clarendon Press, New York, 1995.

[9] L. E. Dubins. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *Amer. J. Math.*, 79:497–516, 1957.

[10] T.O. Espelid. Doubly adaptive quadrature routines based on newton-cote rules. Technical Report 229, Department of Informatics, May 2002.

[11] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design: A practical Guide*. Academic Press, Boston, 1988.

[12] K. Goldberg, D. Halperin, J.-C. Latombe, and R. Wilson, editors. *Algorithmic Foundations of Robotics*. A. K. Peters, Wellesley, MA, 1995.

[13] J. E. Goodman and J. O'Rourke, editors. *Handbook of Discrete and Computational Geometry*. CRC Press LLC, 1997.

[14] Y.K. Hwang and N. Ahuja. Gross motion planning - A survey. *ACM Computing Surveys*, 24(3):219–291, September 1992.

[15] H. Imai and M. Iri. Polygonal approximations of a curve—formulations and algorithms. In G. T. Toussaint, editor, *Computational Morphology*, pages 71–86. North-Holland, Amsterdam, Netherlands, 1988.

[16] P. Jacobs and J. Canny. Planning smooth paths for mobile robots. In Z. Li and J. F. Canny, editors, *Nonholonomic Motion Planning*, pages 271–342. Kluwer Academic Pubishers, 1992.

[17] Y. Kanayama and B. I. Hartman. Smooth local path planning for autonomous vehicles. In *Proc. of the 1989 IEEE International Conference on Robotics and Automation (Vol. 3)*, pages 1265–1270, Scottsdale, AZ, 1989.

[18] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

[19] Luossavaara-Kiirunavaara Aktiebolag (LKAB). http://www.lkab.com.

[20] D. Lutterkort and J. Peters. Smooth paths in a polygonal channel. In *Proceedings of the Conference on Computational Geometry (SCG '99)*, pages 316–321, New York, N.Y., June 13–16 1999. ACM Press.

[21] H. P. Moreton. *Minimum curvature variation curves, networks, and surfaces for fair free-form shape design*. PhD thesis, University of California at Berkeley, 1992.

[22] D. Nieuwenhuisen and M.H. Overmars (2002). Motion planning for camera movements in virtual environments. Technical Report UU-CS-2003-004, Utrecht University: Information and Computing Sciences, Utrecht, the Netherlands, 2003.

[23] U. Reif. Best bounds on the approximation of polynomials and splines by their control structure. *Computer Aided Geometric Design*, 17(6):579–589, 2000.

[24] J.T. Schwartz and M. Sharir. Algorithmic motion planning in robotics. In J. van Leeuwen, editor, *Algorithms and Complexity*, pages 391–430. Elsevier, 1990.

[25] Sandvik Tamrock. *Toro 2500 Electric Technical specification L002500E-1*. Retrieved 16 Dec. 2003 from the World Wide Web: http://www.toro.sandvik.com/.