Initialization Procedure for Radar Target Tracking without Object Movement Constraints

Markus Bühren and Bin Yang
Chair of System Theory and Signal Processing
University of Stuttgart, Germany
www.LSS.uni-stuttgart.de

***Abstract*—The tracking of radar targets in automotive applications often relies on certain constraints to the movement of objects. For example, the objects of interest in an ACC system (adaptive cruise control) are other vehicles that are positioned straight ahead and moving in approximately the same direction as the observer. In this case, a single radar measurement (distance to target, bearing angle and Doppler velocity) contains – neglecting measurement noise – full information about the position (by distance and angle) and the movement state. Thus, the initialization of tracks can be done based on a single measurement.**

Without the mentioned assumption, no information about the movement direction of the object is contained in a single measurement. Theoretically, at least two measurements are necessary to extract information about the object movement direction. But due to severe measurement noise and quantization, even three or more measurements may contain misleading information about the movement state.

In this paper we present an initialization procedure for radar target tracking without any constraints to object movement. In the first cycles of a new track, the state estimation is computed by a linear regression method. After that, the track state is handed over to a Kalman filter which does the tracking for the rest of the track's lifetime.

***Index Terms*—Radar tracking, Radar signal processing, Road vehicle radar**

I. Introduction

In applications of automotive radar of today, certain object movement constraints are made in order to facilitate the tracking of radar targets. The objects of interest in ACC (adaptive cruise control) systems are straight ahead and moving in the same (or nearly the same) direction as the observer. In a parking aid application, only static objects are considered. And in blind spot surveillance or lane departure warning systems, again objects with approximately the same driving direction have to be detected.

If the movement direction of objects is fixed by a certain assumption, the movement state of an object can – aside from measurement noise – be derived from a single radar measurement (distance to target, bearing angle and Doppler velocity). The position of the object relative to the sensor can be computed by distance and bearing angle, while the object speed is given by the combination of own vehicle speed (measured by inertial sensors) and measured Doppler velocity. A situation like that is shown in Fig. 1. The circle marks the detected target position and the attached arrow shows the direction of the measured radial Doppler velocity.

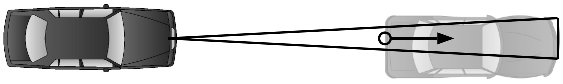

Fig. 1. Typical vehicle-following situation

The given assumptions about object movement facilitate the tracking algorithm. However, objects that do not follow these assumptions, like vehicles crossing the own driving lane or moving perpendicular to the own driving direction, will likely be tracked erroneously or not at all. But in future applications, for example designed to assist the driver in more complex situations, like intersections in urban areas, such movement constraints can not be used anymore.

Without assumptions about the movement direction, the initialization of tracks is a serious problem. In the situation shown in Fig. 2, where a vehicle is crossing the own driving lane, a single radar measurement does not contain full information about the movement state of the object. The measured Doppler velocity reflects only the component of the object speed vector projected on the line from the sensor to the position of the detection.

The Kalman filter is widely used for target tracking [1, 2]. In many applications, the initial state estima-

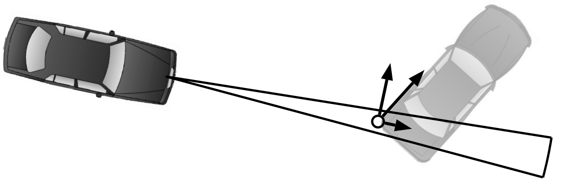

Fig. 2. Object vehicle crossing the own driving lane

tion can directly be derived from the initial measurement. The Kalman filter is often robust against errors in the initial state, but this is not true in our case. We are interested in a precise state estimation after as few cycles as possible, and the initial state has strong influence on the first state estimations. If the object's movement direction is initialized with large error, the tracking filter will very likely diverge. In case of divergence, the track will not receive any new measurements in the data association step. While it may be kept alive during some cycles, the next measurements originating from the same object will be used to form a new track, where the initialization may be better or lead to another diverging track.

In section II below, we will present a method to estimate the target state in the first cycles of a tracks's lifetime by fitting a straight line through the measured positions instead of using the standard Kalman filter initialization procedure. After that, we will show in section III how the state estimation can be handed over to the Kalman filter which then performs the state estimation until the track is deleted. Some results with real radar measurement data are presented in section IV.

II. Initial state estimation

Without measurement noise, the vector that connects the measured positions in two consecutive cycles would exactly represent the movement direction and speed of the object. However, under the influence of measurement noise and quantization, the constructed vector can point into a completely wrong direction, especially if the object is only slightly moving relative to the own vehicle.

As an example, in Fig. 3 a trace of twelve real radar measurements is shown as crosses (some are hidden by others), transformed from the original representation in bearing and distance to cartesian coordinates (note that the scalings in x- and y-direction are different). The observing vehicle, equipped with two Tyco Electronics M/A-Com 24 GHz short range radar sensors as presented in [3], was looking to the right, i.e. in x-direction. An object vehicle was driving perpendicular to the observer, i.e. in negative y-direction. The measurements are severely quantized in bearing with a quantization step of $1°$, thus the measurements seem to be lying on five distinct levels in y-direction.

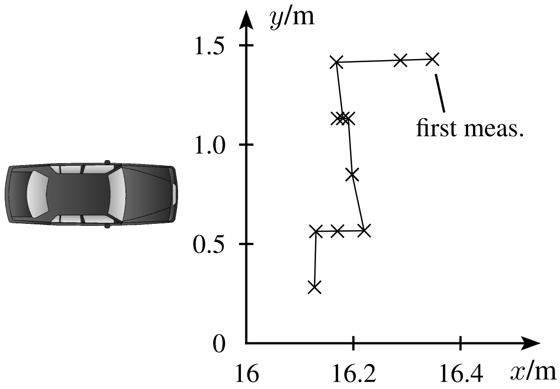

Fig. 3. First detections of vehicle moving from top to bottom

Clearly, the first three measurements on the top, which have the measured bearing angle of $5°$ in common, pretend a movement of the object in direction towards the observer. A Kalman filter, initialized with the movement direction formed by the first two or three measurements, would typically show a significant delay in following the next measurements until the correct movement direction of the object is estimated.

Due to this problem, we propose to use a regression procedure as follows. During the first (up to 10) cycles of a track, a regression line is constructed through the measured target positions. As typical cycle times in automotive radar sensors are around $30 - 50$ ms, the movement during the first cycles can be approximated as being linear. The computation of the motion state is done in two separate steps. First a regression line is computed using a Total-Least-Squares (TLS) approach. After that, estimates for the movement direction, velocity and object position, which is lying on the regression line, are computed.

A. *Regression line computation*

The computation of a regression line using a Total-Least-Squares criterion can be found in the literature. However, we will give the solution to this optimization in detail here, as some readers might not have the solution of the TLS optimization problem in mind.

As shown in Fig. 4, we define a regression line by the two-dimensional vector $\mathbf{p}$, which is perpendicular to the regression line itself and ends on the point of the regression line that is closest to the origin. The task at hand is now to find the vector $\mathbf{p}$ which minimizes the sum of squared distances between the last N measured positions $\mathbf{y}_1, \ldots, \mathbf{y}_N$ and the regression line. The number N increases over time as more measurements are assigned to the current track.

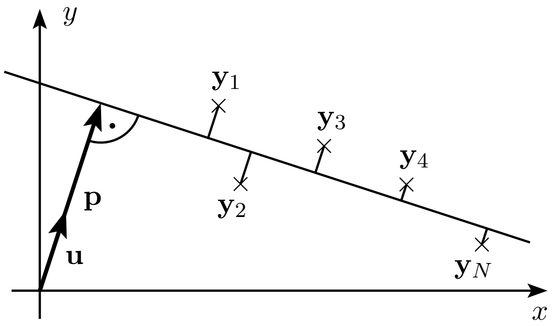

Fig. 4. Regression line computation using Total Least Squares

Let the vector $\mathbf{u}$ be the unit vector in direction of $\mathbf{p}$, i.e. $\mathbf{u} = \mathbf{p}/||\mathbf{p}||$. Then the distance d_n from measurement vector $\mathbf{y}_n$ to the regression line can be obtained as the scalar product between the measurement vector itself and vector $\mathbf{u}$, minus the distance of the regression line from the origin. With $r = ||\mathbf{p}||$, this results in

$$d_n(r, \mathbf{u}) = ||\mathbf{y}_n^T \mathbf{u} - r||. \qquad (1)$$

By gathering all measurements (column vectors) in a $N \times 2$-matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N]^T$, the sum of the squared distances can be written as

$$F(r, \mathbf{u}) = \sum_{n=1}^{N} d_n^2(r, \mathbf{u}) = ||\mathbf{Y}\mathbf{u} - r\mathbf{1}||^2 \qquad (2)$$

with the $N \times 1$-vector $\mathbf{1} = [1, 1, \ldots, 1]^T$. Setting the derivatives of this expression by r and $\mathbf{u}$ to zero results in the following intuitive solution: The estimate $\hat{\mathbf{u}}$ for the vector $\mathbf{u}$ is the eigenvector of the smaller eigenvalue of the 2×2-matrix

$$\mathbf{C}_{\mathbf{xy}} = \begin{bmatrix} \hat{\sigma}_x^2 & \hat{c}_{xy} \\ \hat{c}_{xy} & \hat{\sigma}_y^2 \end{bmatrix}, \qquad (3)$$

where $\hat{\sigma}_x^2$ and $\hat{\sigma}_y^2$ are the sample variances of the x- and y-coordinates of the measurements and $\hat{c}_{xy}$ is the sample covariance between those. The eigenvector to the larger eigenvalue points into the direction of the regression line. Due to the parametrization of the regression line, we here need the second eigenvector that is perpendicular to the regression line.

Note that matrix $\mathbf{C}_{xy}$ can be updated with low computational effort when a new measurement is added in the next cycle. The estimate for the length r of vector $\mathbf{p}$ can finally be computed as the length of the projection of the mean measurement vector (the center of the cloud of points) on the vector $\hat{\mathbf{u}}$, i.e.

$$\hat{r} = \hat{\mathbf{u}}^T \frac{1}{N} \sum_{n=1}^{N} \mathbf{y}_n \qquad (4)$$

B. Speed and position estimation

After computing the regression line, the speed and position of the observed object are estimated. According to Fig. 5, the model for a linear two-dimensional movement can be written as

$$\mathbf{x}_n = \hat{\mathbf{p}} + \hat{\mathbf{u}}^{\perp} \cdot (d_0 + v\, n\, T), \qquad (5)$$

where $\mathbf{x}_n$ is the position at time instance n, the vector $\hat{\mathbf{u}}^{\perp}$ is a unit vector in direction of the estimated regression line, v is the speed to be estimated, T is the cycle time and d_0 specifies the position of the object at time $n = 0$ and has to be estimated as well. The criterion to be minimized is thus

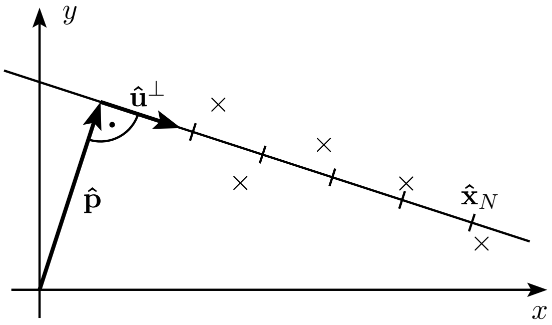

Fig. 5. Estimation of position and speed based on regression line

$$F(d_0, v) = \sum_{n=1}^{N} ||\mathbf{y}_n - \hat{\mathbf{p}} - \hat{\mathbf{u}}^{\perp}(d_0 + n\, v\, T)||^2, \quad (6)$$

where $\mathbf{y}_n$ are again the measurements of the true states $\mathbf{x}_n$. The derivatives of equation (6) by d_0 and v are

$$\frac{\partial F}{\partial d_0} = -2 \sum_{n=1}^{N} \Big((\hat{\mathbf{u}}^{\perp})^T \mathbf{y}_n - (d_0 + n\, v\, T) \Big) \text{ and } \quad (7)$$

$$\frac{\partial F}{\partial v} = -2 \sum_{n=1}^{N} n \Big((\hat{\mathbf{u}}^{\perp})^T \mathbf{y}_n - (d_0 + n\, v\, T) \Big), \qquad (8)$$

where the orthogonality between $\hat{\mathbf{u}}^{\perp}$ and $\hat{\mathbf{p}}$ has been used. Setting both derivatives to zero results in the linear system of equations

$$\begin{bmatrix} N & T \sum_n n \\ \sum_n n & T \sum_n n^2 \end{bmatrix} \begin{bmatrix} d_0 \\ v \end{bmatrix} = \begin{bmatrix} \sum_n \; (\hat{\mathbf{u}}^{\perp})^T \mathbf{y}_n \\ \sum_n n (\hat{\mathbf{u}}^{\perp})^T \mathbf{y}_n \end{bmatrix} \qquad (9)$$

for d_0 and v.

The current object position estimation is finally

$$\hat{\mathbf{x}}_{\mathbf{N}} = \hat{\mathbf{p}} + \hat{\mathbf{u}}^{\perp} \cdot (\hat{d}_0 + \hat{v}\, N\, T) \tag{10}$$

with the estimates $\hat{d}_0$ and $\hat{v}$. This expression needs to be evaluated only once. As we used the index n to number the cycles in the tracks's lifetime, index $n = N$ corresponds to the current cycle. The position estimates for past time instances are not needed and likely to change when a new measurement is added in the next cycle. If, however, no new measurement is assigned to the current track, the same estimates can be used to extrapolate the state estimation. In this case, equation (10) has to be evaluated for $N + 1$.

Note that the estimation of movement direction, speed and position can also be jointly formulated as one optimization problem, instead of two consecutive optimizations as described before. The solution will be slightly different, but the resulting optimization problem will be nonlinear and most probably only to be solved iteratively, which is not favorable in a real-time system.

C. *Computational effort*

The regression procedure is computationally not very demanding. The update of matrix $\mathbf{C}_{xy}$ from equation (3) requires less than ten additions and ten multiplications. The eigenvalue decomposition of a 2×2-matrix can be done analytically and also requires only a low number of operations. For the evaluation of equation (4), intermediate values from the update of matrix $\mathbf{C}_{xy}$ can again be exploited and only few more operations are needed.

The inverse of the matrix on the left side of equation (9) can be precomputed, as the number of different values of N is limited. The most computational effort arises in the computation of the right side of the equation, as the vector $\hat{\mathbf{u}}^{\perp}$ changes in each iteration. Here, around $3N$ multiplications and $2N$ additions have to performed. But as the maximum number of N is usually less than or equal 10, the whole computational effort for the regression line computation and the position and speed estimation is reasonably low.

D. *Selection of the coordinate system*

The regression can be done either in the sensor coordinate system or in a global coordinate system, where both the own vehicle and the object are moving through. In both cases it might happen that all measurements gather around one point. When working in the observer's coordinate system, this occurs if observer and object are moving in the same direction with the same speed; when using a global coordinate system, the measurements of fixed objects are expected to cumulate at a single point. In both cases, the computation of a regression line does not make sense, as its direction will randomly change with each new measurement incorporated and the estimated speed v will be close to zero.

In order to save the computational effort for the regression procedure, we first determine if the distance between any measured position and the average of all measured positions exceeds a certain threshold. The threshold depends on the coordinate system in use. When computing the regression line in the sensor coordinate system, our experiments show good performance if the regression procedure is started after either the bearing angle has changed about at least 3° or the measured distance has changed about 20 cm. If the threshold is not exceeded, the position estimate in the selected coordinate system is set to the average of all positions and the speed is set to zero.

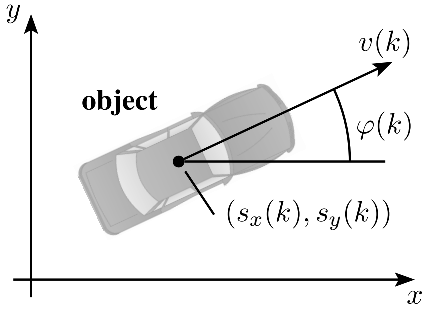

Fig. 6. Object state variables in global coordinate system

III. Handover to Kalman-filter

After the number of measurements N associated to the given track reaches a given limit (for example 10 as stated before), the state estimation task shall be handed over from the regression procedure to the Kalman filter. In the initialization of a Kalman filter, four things have to be set: the input (or driving noise) covariance matrix, the measurement noise covariance matrix, the initial state and the initial state covariance matrix. The first two elements can be set in a standard way and are not affected by particular initialization conditions. For the latter two, we first take a short look at the Kalman filter motion model we are using. It is described in detail in [4] and [5]. The state vector $\mathbf{x}(k)$ is defined as

$$\mathbf{x}(k) = \left[s_x(k)\ s_y(k)\ v(k)\ \varphi(k)\ a(k)\ \delta(k)\right]^T, \tag{11}$$

where $(s_x(k), s_y(k))$ is the global vehicle position, $v(k)$ and $\varphi(k)$ are tangential speed and movement direction (see Fig. 6) and $a(k)$ and $\delta(k)$ represent the acceleration and steering wheel angle.

The initialization of the state estimation is straightforward from the regression line computation introduced before. The initial position $(s_x(k), s_y(k))$ is set to the result of equation (10), while the object velocity $v(k)$ was computed by solving the system of equations (9). The driving direction $\varphi(k)$ can be obtained using the $\arctan$ function on the regression line parameters. About the last two state variables, the acceleration $a(k)$ and the steering wheel angle $\delta(k)$, no information is available, not even through the regression line procedure. We set these values to 0 and leave the task of estimating them to the Kalman filter. In the further development of our tracking system, currently we are investigating if the acceleration and/or the steering angle can be left away without reducing the performance.

The initialization of the state covariance matrix, in contrast, requires to take sensor characteristics into account. In most automotive radar sensors, the distance can be measured quite precisely, while the bearing angle is subject to considerable measurement noise and quantization. This means that the initial estimate for the x-position – relative to the sensor – of an object straight ahead is better than the estimated relative y-position. Further, in a vehicle-following situation as shown in Fig. 1, the tangential speed estimation will be better than the estimate for the driving direction. The opposite is true if an object is moving perpendicular to the own driving direction.

Next we will express these qualitative statements in numbers. The desired way would be to directly compute the variances and covariances of all state variable estimates. However, the computation of these terms is very complicated, as an eigenvalue decomposition and nonlinear functions are involved. Due to this, we do some heuristical approximations for the values we want to find.

For the computation of an approximation for the variances of and covariance between $x(k)$ and $y(k)$, we consider the measurement variances in distance and bearing and compute the before mentioned values for the simplified situation of estimating the position in cartesian coordinates from a single radar measurement. We define a random variable R for the distance measurements with the expected (true) value $\mathrm{E}(R) = \bar{R}$ and variance $\mathrm{Var}(R) = \sigma_R^2$. Further, let α be a random variable describing the bearing angle measurements, with $\mathrm{E}(\alpha) = \bar{\alpha}$ and $\mathrm{Var}(\alpha) = \sigma_\alpha^2$. The transformation from polar coordinates (R, α) to cartesian coordinates (x, y) is

$$\begin{bmatrix} x \\ y \end{bmatrix} = R \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} \approx R \begin{bmatrix} \cos(\bar{\alpha}) - (\alpha - \bar{\alpha}) \sin(\bar{\alpha}) \\ \sin(\bar{\alpha}) + (\alpha - \bar{\alpha}) \cos(\bar{\alpha}) \end{bmatrix}. \tag{12}$$

We will now use this approximation in order to compute an approximation for the covariance matrix $\mathbf{C}_{xy}$ of the vector $[x, y]^T$. With

$$\mathrm{E} \begin{bmatrix} x \\ y \end{bmatrix} \approx \bar{R} \begin{bmatrix} \cos(\bar{\alpha}) \\ \sin(\bar{\alpha}) \end{bmatrix} \tag{13}$$

we get

$$\begin{bmatrix} x - \mathrm{E}\, x \\ y - \mathrm{E}\, y \end{bmatrix} \approx \begin{bmatrix} (R - \bar{R}) \cos(\bar{\alpha}) - R(\alpha - \bar{\alpha}) \sin(\bar{\alpha}) \\ (R - \bar{R}) \sin(\bar{\alpha}) + R(\alpha - \bar{\alpha}) \cos(\bar{\alpha}) \end{bmatrix}. \tag{14}$$

The covariance matrix is

$$\mathbf{C}_{xy} = \begin{bmatrix} \mathrm{Var}(x) & \mathrm{Cov}(x, y) \\ \mathrm{Cov}(x, y) & \mathrm{Var}(y) \end{bmatrix} \tag{15}$$

Using equation (14) and the assumption that R and α are statistically independent, we can now compute the elements of the covariance matrix:

$$\mathrm{Var}(x) \approx \sigma_R^2 \cos^2(\bar{\alpha}) + (\sigma_R^2 + \bar{R}^2)\sigma_\alpha^2 \sin^2(\bar{\alpha}) \tag{16}$$

$$\mathrm{Var}(y) \approx \sigma_R^2 \sin^2(\bar{\alpha}) + (\sigma_R^2 + \bar{R}^2)\sigma_\alpha^2 \cos^2(\bar{\alpha}) \tag{17}$$

If an object is located on the x-axis ($\bar{\alpha} = 0$), the variance of x is equal to the variance of R. The variance of y is then $\sigma_y^2 = (\sigma_R^2 + \bar{R})\sigma_\alpha^2$; as expected it will rise if the distance to the target increases. Finally, the covariance between x and y results in

$$\mathrm{Cov}(x, y) \approx \tfrac{1}{2} \left(\sigma_R^2 - (\sigma_R^2 + \bar{R}^2)\sigma_\alpha^2 \right) \sin(2\bar{\alpha}). \tag{18}$$

The covariance is equal to zero if the target is located on the x-axis, as here the line of constant bearing angle and the tangents on the circles of constant radius are parallel to the x- and y-axes.

We have now computed the covariance matrix of the position vector given the true distance $\bar{R}$ and the true bearing angle $\bar{\alpha}$. As we do not know these values, we use the measurement values from the target list instead. The involved sine- and cosine functions will not change dramatically if the angle measurement deviates from the true value by some degrees, so we will get a reasonable approximation. The measurement variances σ_R^2 and σ_α^2 finally are depending on the sensor characteristics and the target RCS. For example, with the near range sensors we are using and a vehicle in a distance of 15 m, we calculate with the standard deviations $\sigma_R = 2\,\mathrm{cm}$ and $\sigma_\alpha = 0.75°$. However, the possible range for these values in quite large.

The computed covariance matrix, as said before, corresponds to the situation where we use a single measurement to estimate the object position. The coordinate system our derivations have been done in was the cartesian coordinate system of the sensor. In order to get into the global coordinate system that we are using in our tracking model, only a rotation of this matrix about the angle between the global and the sensor's x-axis is necessary. Further, in the regression line procedure, we used N consecutive measurements to estimate the position. Thus the rotated covariance matrix has to be divided by the number of measurements N before it can be set as the upper-left 2×2-submatrix of our 6×6 state covariance matrix.

The derivation of the variances of the estimated driving direction and relative speed is even more complicated than for the position in x and y. Heuristically tuned, fixed parameters allow good results in the most cases, but we are currently also investigating possible approximations for the missing variances. At the time, we propose to set all values except the upper-left 2×2-matrix and the lower four diagonal elements to zero. As the remaining parameters of the state covariance matrix are subject to change during further development of our tracking system, we do not give numerical values here.

IV. Results

In this section, we will present the result of the regression line computation using the data already shown in Fig. 3. Even if we proposed not to compute the regression line before the measured positions leave a certain region, for generating this real data example we started the regression already with the second measurement. The position estimations are marked in Fig. 7 as circles, while the measurements are again shown as crosses. Note that we exchanged the axes in this figure in order to allow a larger display. The first measurement is located in the upper right corner in this figure.

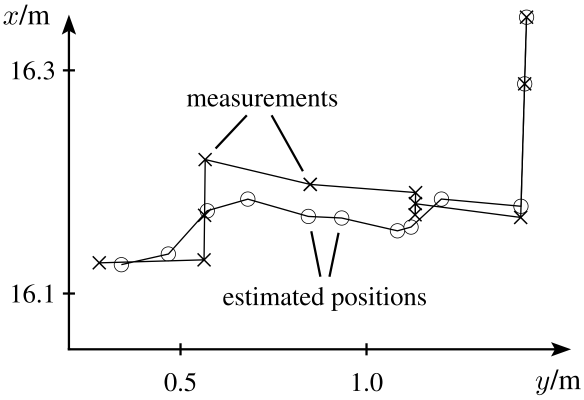

Fig. 7. Position estimations using regression line procedure

The robust smoothing effect of the regression line computation can well be observed. While the measurements "jump" due to the quantization of the bearing angle, the trajectory of the position estimations is a good estimation for the true object trajectory.

A comparison of our proposed regression initialization with a direct Kalman filter initialization would be straightforward. However, as the performance of a Kalman filter depends on many different parameters, we have omitted such a comparison here.

V. Conclusion

In this paper, we have discussed the problems arising in the initialization of a Kalman filter for radar target tracking in automotive applications. A single radar measurement is not sufficient to initialize the state of a Kalman filter, as it does not contain full information about the movement direction of the observed object. We introduced an initialization procedure that is based on a regression line through the measured object positions in the first cycles of a track's lifetime. A favorable smoothing effect on the measurement positions was shown by results using real measurement data.

After some cycles, sufficient information about the movement state, including the movement direction and speed, was gathered. The result of the regression procedure is then used to initialize a Kalman filter. The initial covariance matrix is computed under consideration of the different measurement variances in distance and bearing angle.

References

[1] Samuel S. Blackman, *Multiple-Target Tracking with Radar Applications*, Artech House, 1986.

[2] Eli Brookner, *Tracking and Kalman Filtering made Easy*, John Wiley & Sons, New York, 1998.

[3] Hermann Henftling, Dirk Klotzbücher, and Christian Frank, "Ultra wide band 24GHz sequential lobing radar for automotive applications," in *Proc. Intern. Radar Symposium (IRS)*, Berlin, Germany, Sept. 2005, pp. 79–82.

[4] Markus Bühren and Bin Yang, "On motion models for target tracking in automotive applications," in *Proc. Intern. Workshop on Intelligent Transportation (WIT)*, Hamburg, Germany, Mar. 2007.

[5] Markus Bühren and Bin Yang, "A global motion model for target tracking in automotive applications," in *Proc. IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, Hawaii, USA, Apr. 2007, vol. 2, pp. 313–316.