Community Map Generation based on Trace-Collection for GNSS Outdoor and RF-based Indoor Localization Applications

Brian Niehöfer *, Andreas Lewandowski *, Ralf Burda *, Christian Wietfeld *, Franziskus Bauer† and Oliver Lüert‡
*Communication Networks Institute
Dortmund, University of Technology, Germany
{brian.niehoefer, andreas.lewandowski, ralf.burda, christian.wietfeld}@tu-dortmund.de
†Ford of Europe Product Development, Cologne, Germany
fbauer6@ford.com
‡mimoOn GmbH, Duisburg, Germany
oliver.luert@mimoon.de

***Abstract*—The paper describes a map generation system, which relies on random individual contributions of GNSS (Global Navigation Satellite Systems) traced movements for outdoor scenarios and traces from a RF-based indoor localization system. In a typical use case, mobile phone users would join a specific community to contribute their movements along streets, roads or pathways in form of so called journeys. The proposed algorithms presented in this paper are also able to generate a precise map for indoor scenarios, with which certain changes the accuracy of the RF localization system can be mitigated. Neither contributing subscribers nor the map generation need to have an á priori knowledge of the charted area. The approach presented here comprises the journey recording, the upload process to a common server and the processing algorithm for map generation. The filter mechanisms and adaptive plausibility checks applied to the raw data are key components for the deduction of precise street and movement maps. Summing up, the paper provides a proof of concept of a map generation system for different in- and outdoor application scenarios relying on random individual movements to demonstrate the performance of the algorithmic approach and the adaptablity of the map generator.**

***Keywords*-map generation; community map; street-attribute creation; RF-based indoor localization.**

I. Introduction

Recently GNSS (Global Navigation Satellite System) receivers have become a standard feature of business class mobile phones. Thus, a new and rapidly growing user group is employing terrestrial navigation and merges with the already proliferating market of vehicle based and portable navigation systems [7][8]. Assessing system performance by traits as ”being up to date”, ”good accuracy” and ”availability of annotations”, traditional GPS navigation systems always rely on already out–dated map data and thus are always lagging behind. Moreover they are limited in their resolution by local map storage capacity. In contrast community based map collection and generation scales with the number of members and is envisioned to overcome these limitations. In addition, memory demand on the mobile user side may be lowered as the used and locally stored map area at some time only need to be a subset of all available maps at the expense of increasing data volumes being exchanged. The dynamic upload of requested map data implicitly enables the use of up to date information. From the market perspective, such an approach must ensure an additional value for potential users in order to be accepted, e.g., in terms of accuracy, pricing or available annotations to points of interest [3][4]. Potential user groups range from pedestrians, joggers or hikers to car drivers. The performance indicators, as mentioned above, are assumed to match this requirement and in addition enhance coverage (i.e., in rural and retired areas). As GPS or other navigation resources such as GALILEO [13] are expected to be integrated into even low budget devices soon, the potential number of subscribers to a community based system is likely to rise. The minimum number of potential, technology enabling subscribers, therefore is supposed to be already active. Thus the authors strongly assume that an operational system, which follows the algorithms proposed in this work can be established in the current market situation.

Another emerging technology field is the application of RF-based indoor localization systems (e.g., in safety critical scenarios). In comparison to [1], this paper shows further extensions to the map generation algorithm addressing the specific conditions of an indoor positioning system. In comparison to GNSS systems, RF-based localization systems are highly vulnerable to changes in the environment caused by moving persons or any relocations of inventory. RF indoor localization relies on the evaluation of the Received Signal Strength Indicator (RSSI), a Time of Arrival (ToA) measurement or hybrid approaches. As typical application areas are highly influenced by multipath fading effects, the accuracy of position estimation is varying over the time. Considering a high number of journeys collected in these environments, temporal signal fluctuations caused by specific conditions can be mitigated using the map generation framework. Once a sophisticated user map is generated, the position estimation can be supported by the generated map during normal operations. In case of changes in the traffic infrastructure, new journeys are detected by persitent deviation and consequently lead to map updates. This process is envisioned to also support the detection of temporal situations like blockings due to accidents or construction sites.

Parameters of the depicted algorithm used in this paper have been deduced by pure heuristics. It should be noted that the goal of this work has been to prove the concept of a reasonable solution for distributed map processing rather than finding an optimum. The authors want to emphasize that the key contribution of this paper is the architecture of the novel, community based approach, especially for indoor scenarios. Furthermore, performance evaluations for motorway, city and pedestrian scenarios for GNSS journeys are shown.

In the sequel Section II describes the overall system to record journeys and to transfer them to a server. The section also addresses the processing stages of map merge and map storage. Section III gives details on the particular map generation algorithms while pointing out quantified indicators for proper map performance. Section IV describes the scenarios, which have been applied for performance evaluation. Section V provides results, explaining exemplary maps and presents quantitative measures. It is assumed that an increased size of the set of journeys including a particular route is sufficient to guarantee convergence of the traced data to a street map of reasonable quality. Section VI wraps up the major findings in a conclusion and states an outlook to further work.

II. System Overview

The architecture of the proposed map generation system is flat and operates in a client server fashion. It is based on a large number of mobile *trace recording devices (TRDs)*, e.g., a group of *collaborative navigation devices (CND)* or, in case of indoor scenarios, corresponding *indoor localization tags (ILTs)*. Furthermore one *central server or map generation server (MGS)* is necessary to accomplish the map generation process. In the first place, the MGS is responsible for filtering the raw traced data as well as for generating and storing the resulting map. In addition, it will keep track of a limited trace history for plausibility checks and finally hadles the task of scheduling the distribution of requested results back to the CNDs. It should be noted that in a practical use case TRD and CND are likely to be co–located or even different functions of the same physical device, although this is not a prerequisite for successful operation.

Figure 1 provides an overview of the community based map generation system. It shows the graphical user interface of a sample collaborative navigation device (community user) and the server providing databases for the recorded journeys and the resulting map. A further task of the server is the cumulation of the trace data into a concise map, which can be returned to the systems users.

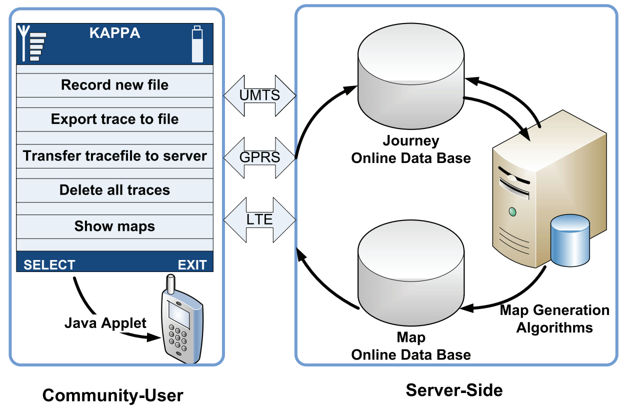

Figure 1. System Overview

A. Collaborative Navigation Device (CND)

The used CND is a simple GPS equipped mobile phone, which has been used as a trace recorder. It executes a Java-Midlet and stores the recorded waypoints as list of journeys (see Figure 2). Each journey is defined by an arbitrary

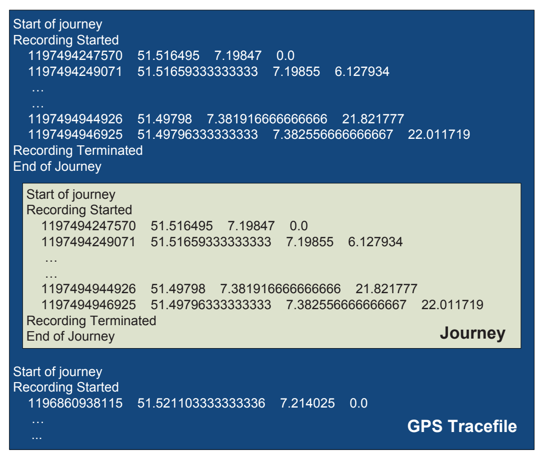

Figure 2. Sample Trace Data

number (>= 2) of waypoint samples. The first and the last sample denot the starting point of the journey and the ending point respectively. For the findings in this paper an initial setup has been chosen, which samples waypoints and records them in time intervals of 1 second. Each waypoint summarizes the geographical position in latitude and longitude along with an UTC time stamp. Altitude information is omitted due to the poor GPS resolution in this respect. Further annotation of the location date is possible in general but has been omitted for clarity as it would not support the concepts focused here. Figure 2 shows an exemplary part of a GPS trace file, whereby the first number in each row resembles the time stamp, the second and third the latitude respective longitude position and the forth the calculated velocity for the past two waypoints. We calculate the velocity during the recording of the GPS data to save computing time at the server side. Note that each GPS Trace File is defined as a container only and is used to transmit the relevant data from the mobile to the MGS. For this reason those text files are always created after each upload request of the CND.

The TRD is different in the reference design for the indoor localization system. Here the data is captured and analyzed by a central localization server, whereby an ILT sends the current position directly over the air to the server. This setup has been chosen due to the limited resources of processing capabilty and battery lifetime in the TRDs.

B. Map Storage Architecture

Figure 3 depicts the *object flow chart (OFC)* diagram for all data objects from raw journey data, summarized in GPS Trace Files as explained in II-A, to a final map knowledge representation. To facilitate the association of journey data to the final map, another storage entity is introduced, which will be called a *segment* throughout this paper. Each segment consists of a processed version of the former waypoints. In a first step, all recorded waypoints have to pass at least three redundancy checks, to minimize the effect of possible erroneous measurements in the GPS tracking process, before the connecting edges are accepted as a segment. Afterwards each segment is expanded by some additional information like an unique ID or the average velocity on a specified segment part, but this will be explained in Section III-D in detail. All segments are assumed to represent a potential path in the final map. The differences between a segment S and a journey J are the eliminated illogical waypoints W by using Consistency Checks. So (S) is an annotated subset of $\bigcup J$. (This will be will be explained in more detail in Section III-A. Each segment $S_y = \{N_1, N_2, \cdots N_n\}$ is made up by an ordered sequence of nodes N. The set of segments $\mathbb{S}$ comprises all nodes:

$$\bigcap_x S_x = \mathbb{N} \quad \text{with} \quad S_x \in \mathbb{S} \tag{1}$$

finally constitutes the points, which are known to the map. The connections between 2 or more segments are also special nodes called a *Crossing* (C), which are defined by

$$\exists (x, y) \rightarrow \mathrm{C} \in S_x \wedge \mathrm{C} \in S_y. \tag{2}$$

The relation $\mathbb{C} \subseteq \mathbb{N}$ is self evident. Each crossing C in turn may be entered via so called incoming segments $\mathbb{I}_C \subset \mathbb{S}$ and may be left via exiting segments $\mathbb{E}_C \subset \mathbb{S}$. In the general case $\mathbb{E}_C \bigcup \mathbb{I}_C = \varnothing$ is allowed. Crossings and the respective input and output subsets are stored in a so called Segment-Transition-Table.

The mentioned segment attributes are relevant for the plausibility check and intelligent filters. Furthermore, a source table stores the history of a segment with the underlying journeys, which were used to retrieve the particular

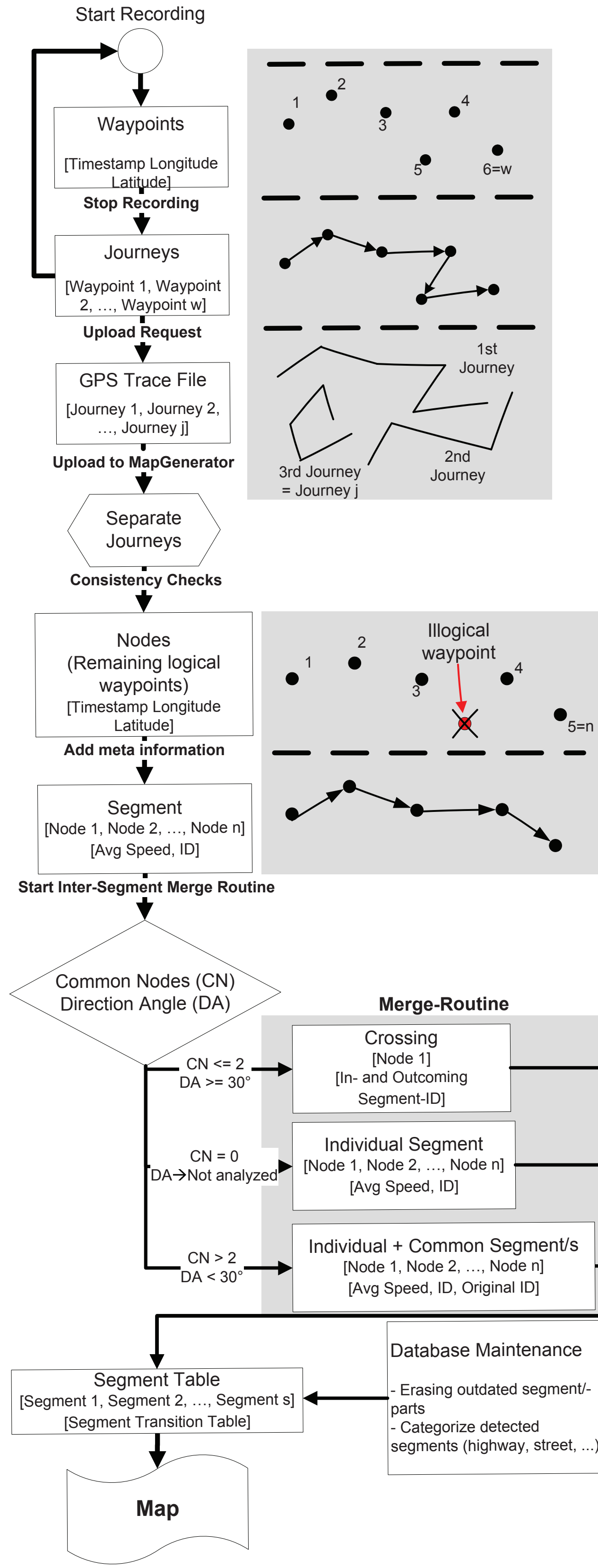

Figure 3. Object flow chart(OFC) diagram for map knowledge

segment. This enables an a-posteriori correction of the generated map in case subscribers or individual journeys turn out not to be trustworthy.

III. MAP GENERATION ALGORITHMS

In a first step, we collect journeys from different subscribers and analyze them. In order to retrieve a realistic map constellation, it is also necessary to analyze every trace for potential duplicates of journeys, say the way to work on different days. In general all journey data is subject to noise due to positioning inaccuracy. The major task of the pre-filter for the map generation is the detection and elimination of duplicates by means of an intelligent reasoning in order to yield a realistic graphical display.These filter algorithms are described in the following.

Presently, all filter algorithms are operating at the server side, although it would be possible to export some to the TRD, like the upcoming *Intra Journey Merge Routine (IJMR)*. To keep the algorithm-structure clearly and transparent, we decide to defer this aim up to the end in our implementation plan for the future.

Currently all waypoints and recorded journeys may contain evident errors caused by measurement errors from the CND. Therefore, the GPS annotated journeys are checked for inconsistencies. Second, each journey is investigated for multiple passings of the same street. Third the journeys are compared to the existing map data (created by merging already uploaded journeys), the distances are calculated and, where appropriate, journey data is merged with the map data. Finally, the information in the journey data is processed to add additional information to the segment- and the transition-table of the map. Note that all the assumptions for validity and the thresholds for taking actions on the journey data are purely heuristic at this stage of research. They have been retrieved and validated to be useful by a lot of experiments.

A. Consistency Checks

As a first step of validation each trace is checked for obvious inconsistencies. This check starts with a pre-run, which executes a 'Minimum distance between points check'.

Due to the inaccuracy of public GPS or RF-based indoor localization, all waypoints, which are too close ($|W_i - W_{i+1}| < \text{T[m]}$) to a given intermediate starting point are averaged and replaced by the result. T is assumed to be a valuable threshold for different application scenario and localization system. In outdoor scenarios, the averaging process is stopped e.g., when

$$\frac{1}{w}\sum_{l=1}^{w}|W_l - W_{w+1}| \geq 5[\text{m}]. \quad (3)$$

Once this spatial noise filter has been applied, the remaining annotated journey data is subject to three further checks (exemplary parameterized for a motorway scenario) .

1) **Speed:** The speed calculated from time and spatial distance between two recorded points may not be larger than $200\frac{km}{h}$. If this is the case, the second point is assumed to be faulty and discarded.
2) **Acceleration:** Similarly, the acceleration along a journey is checked. If the acceleration is larger than $4\,\frac{m}{s^2}$, the point, which was reached through this unlikely acceleration is discarded.
3) **Direction change:** The angle between two successive segment fragment vectors may not be higher than a certain speed-dependent angle. If this is witnessed in the journey the end point of the second vector is discarded.

The filter parameters are adapted to the use case scenario, as different mobility patterns for indoor and outdoor scenarios are expected. If a waypoint is marked by one of these consistency checks, it is discarded from the journey to smoothen the resulting way. The check is then re-run until no further eliminations have been executed. An example for the direction plausibility test is given Figure 4.

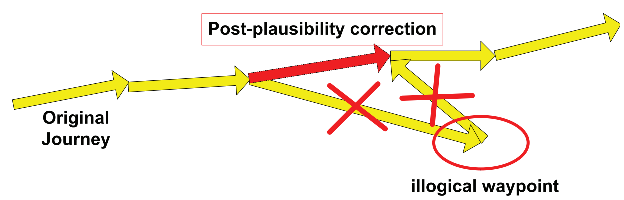

Figure 4. Effects of maximum angle check

The spatial points, which remain in the filtered output of this processing are now called *nodes* N. Due to the averaging process and the other checks $\exists i \rightarrow N_i \ni \mathbb{W}$ is possible. Note that $|\mathbb{W}| \geq |\mathbb{N}|$ holds true.

B. Intra Journey Merge Routine (IJMR)

The upcoming merge routine only controls complete segments. Therefore, redundancy has to be eliminated by checking each journey for intra merge parts. Thus, only relevant waypoints are passed to the segment processing on the server. Redundancy is assumed when a subscriber drives the same street, or a part of it, multiple times per journey, e.g., while searching for a parking lot. The merge routine compares the distance between each fifth (chosen arbitrarily to return processing load) segment fragment of the journey. When the distance decreases below a certain threshold, the routine cuts this part and replaces it by artificial but concise waypoints attributed by new index numbers, directions of travel and time stamps. The processing is then repeated. If none of the 3 consistency criteria is violated, the program passes the result to the segment generation and continues to analyze the next journey. The dissection of the journey into multiple segments enables the reuse of the algorithm for the upcoming *multi journey merge routine (MJMR)*. Both merge

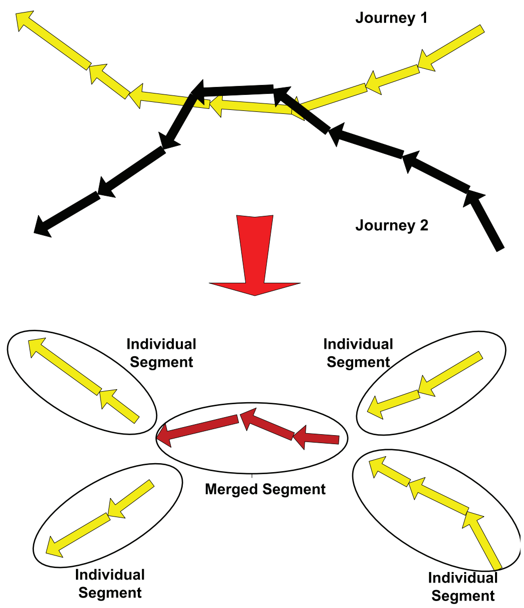

Figure 5. Trace split/merge to segments

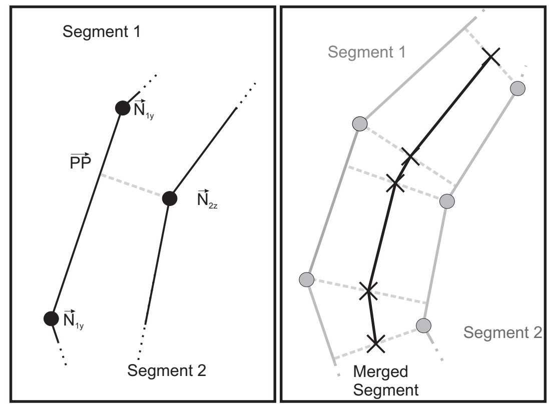

Figure 6. Perpendicular segment alignment

steps are, at present, executed on the server to maintain a concise processing engine. Deployment of the IJMR to the CND or ILT results in a proportional scaling of processing power and journey count in a productive system.

C. Perpendicular Calculation

In a first attempt we compare the distance of all nodes of a new segment record to all nodes already stored in the database. In case of being under a given threshold T (e.g., 13m for GPS outdoor scenarios), the segments are supposed to be copies of already known ones. Moreover, distances between a node and the connecting line between close neighbors may meet the same criterion as being shown in Figure 6-left. To include this case in our algorithm, we use a point–to–point–distance calculation to the closest neighbor node and the line–to–point distance between the predecessor and successor. Now it is possible to calculate the minimum distance to a line, by using the formula displayed below.

$$\left(\vec{N_1} - \vec{N_3} - x \cdot (\vec{N_2} - \vec{N_1})\right) \cdot (\vec{N_2} - \vec{N_1}) = 0 \quad (4)$$

$$\vec{N} = \vec{N_1} + x \cdot (\vec{N_2} - \vec{N_1}) \quad (5)$$

By calculating the 'x' value for which the dot product becomes zero, we draw the perpendicular from the $\vec{N_3}$ in the detail pane of Figure 6-right to the track between $\vec{N_1}$ and $\vec{N_2}$. Obviously, this is the minimum distance between the point of one journey and the segment fragment of the other. Now we can decide precisely, which segments or parts of them (fragments) are below T and feed them to the merge process. Each perpendicular minimum will then be used to increase the reliability factor α of a merged waypoint, which can be understood as an attribute of 'trust' in the correctness of this point. Above a particular numerical threshold of α a segment (either genuine or merged) is consolidated into the final map. For the further calucation steps no distinction is made with respect to the origin of a (now qualified) segment. The weight of a segment in the map is monitored constantly with reference to the count of supporting decisions from new segments or their fragments. The reliability is used as a priority factor in further calculations, like shown in Formula 6, where α is the reliability factor.

$$\vec{N}_{final} = \frac{\alpha_1 \vec{N}_{new} + \alpha_2 \vec{N}_{old}}{\alpha_1 + \alpha_2} \quad (6)$$

In case of a new segment, the reliability factor $\alpha_1 = 1$. Superior segment priority invalidates new *close by* segment fragments, which are still too close to make up for a new segment or street. Aged (not reconfirmed) segments are subject to ageing and consequently will be deleted by a database maintenance routine eventually. Merging two segments of equal α in this perpendicular way leads up to twice the number of estimated nodes than pure averaging as shown in the right pane of Figure 6. The increased node density enables a smooth fitting even to curved trajectories and thus is more realistic. The outer circles in Figure 6-right account for the nodes of the two segment fragments, which we are going to merge in order to eliminate the inevitable noise in position estimation. Concerning our perpendicular calculation (inner lines and circles), we are able to reach highly concise results, as shown by the boundary lines.

By doing so, Figure 7 shows an iterative evolution of the community map (CM) for four exemplary journeys, whereby the dashed lines denote the new added one.

D. Street segment attributes

After inserting new or merged segments into the map, non spatial information is extracted from the segment. Obvious information is the average and maximum velocity (AV, MV) measured across a particular segment fragment and therefore

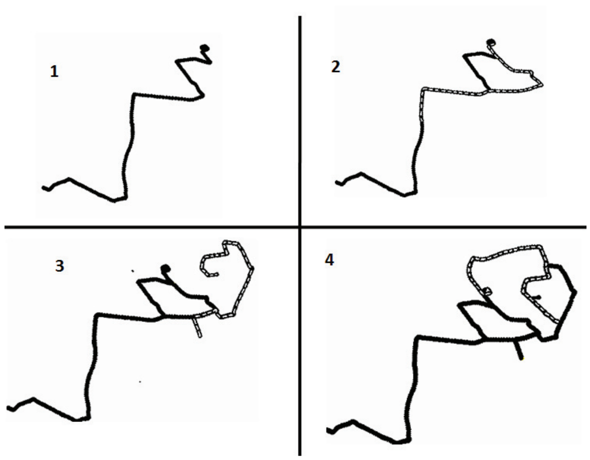

Figure 7. Iterative map evolution from 4 journeys

across the segment it has been associated to. AV then differentiates between highways, streets and walkways in our model, based on characteristics like average speed, direction changes, etc.. Such an approach is useful to deduce different streets, which are directly in parallel, e.g., exit or entry lanes on highways. Further attributes like the time-of-day or day-in-week including the respective AV and MV values per hour are already stored but still remain unused in the algorithms. This information for post analysis is valuable in case the optimization goal of navigation systems employing the CM is minimum delay rather than minimum distance. Community users may update all these attributes by real-time measurements. Taking into account age and first order derivate in case of changing attributes, navigation software gains superior capabilities to detect temporal changes in traffic situations (road work, traffic jam, detours) in nearly real–time.

E. Extensions for new application scenarios

For applying the map generator to indoor scenarios, several changes are implemented to adapt the changed mobility patterns and the different positioning accuracies. The consistency checks have been adapted for the following *movement-classes*:

1) High speed (outdoor)
2) Vehicular (outdoor)
3) Low speed (outdoor)
4) Ultra low speed (indoor)

The degree of freedom for the movement paths is enlarged with every step by reducing the strictness of rules. In the special case of indoor scenarios, we implemented a double stage process, as high fluctuations in the RF-based localization system are observed:

1) Erasing outliers with a coarse filter
2) Adaptation of remaining illogical waypoints by perpendicular calculation

The merge routines have been adapted in terms of dimension and degree of freedom. Additionally, we introduced a speed depended smoothing of the estimated positions.

IV. Scenarios for Performance Evaluation

In order to show the adaptivity and performance of the map generator framework for different scenarios, we chose a top-down approach in terms of scenario size. At first, a classic motorway scenario is evaluated, before showing a city traffic scenario. These are the classic application scenarios for today's navigation devices. But hence, pedestrian mobility in combination with navigation is opening new application fields. Therefore, we provide two scenarios. One outdoor scenario, where the position estimation is accomplished by GNSS, and one indoor scenario, where the position estimation is accomplished by a RF-based localization system.

A. Motorway Scenario

The journeys for the classical Motorway Scenario have been generated in Bochum, Germany (GPS-coordinates approximately: 51.516 N, 7.206 E) at a motorway cross of the frequently driven A40 and A43. Average speeds of around 80 km/h are expected, as the area is mostly speed limited due to heavy commuter traffic. The street structure is characterized by wide spaces and a straight street course.

B. City Traffic

City Traffic is different in terms of speed and position changes compared to motorway scenarios. We have generated GPS journeys again in Bochum. The average speeds are expected to be at around 30 km/h. An interesting point will be how reliable the map generation is working in that application field, as environmental influences like high buildings affect the accuracy of GPS positioning. As the structure of city streets is more fine grained compared to motorways, it is more challenging to generate a highly accurate CM.

C. Pedestrian Movement

In order to evaluate the accuracy of the map generator in pedestrian scenarios, we traced several journeys on our campus on the classical route from out institute to the refectory (Mensa). The journey takes about 3 minutes over a distance of 350m. This scenario is challenging as surrounding buildings affect the accuracy of GNSS position estimation. We deployed an additional tracking application on the iPhone as CND, as we need fast acquisition of the initial user position. In our case this has been accomplished by utilizing the cellular network information to improve the startup performance of the used GNSS system (GPS). This scenario has been chosen to show the capability of adaptation of the map generator framework for low speed and dense position data.

D. *Indoor Localization Testbed*

RF-based indoor localization is a challenging research topic, as the radio channel is prone to interferences caused by fixed infrastructure, other mobile users and the CMD itself. Hence, the position estimation is highly susceptible to these effects, when relying on the common techniques like Received Signal Strength Indicator (RSSI) or Time of Arrival (ToA) range estimations. When thinking of a target environment like an industrial application scenario, employees are expected to walk over commonly used movement paths, as the degree of freedom is usually restricted by constructional circumstances. Thus, the map generator approach constricts the parameter space of journeys by a system calibration time, where external influences are strictly avoided. Hence, during normal operation, external influences are mitigated as the position estimation is corrected to the map location.

In the context of this paper, it is very interesting to learn about the scalability of the map generator in terms of size of scenario and movement behavior. We analyse a mobile indoor localization testbed, consisting of a model train as depicted in Figure 8.

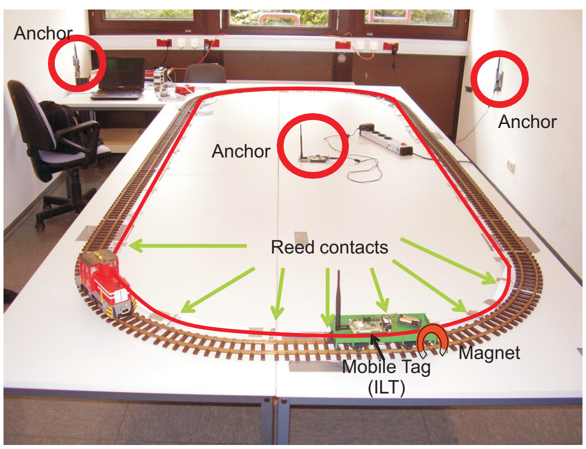

Figure 8. Indoor localization testbed for performance evaluation of the map generator

A so called Chirp Spread Spectrum (CSS) [12] localization system is applied, which uses ToA for range estimation. This system is an extension of the well known IEEE802.15.4 standard [9] and is standardized in the extension IEEE802.15.4a [10] as an alternative Physical Layer (PHY) implementation. We installed 5 fixed anchor nodes and placed an ILT on the model train. The localization error is expected to be at a maximum of 2 meters using this constellation. This is obviously too high for such a small scale scenario. Therefore, the generation of a movement map could, in this case, enhance the localization accuracy and reduce the number of position measurements, as the position change can be predicted based on the current speed of the user.

V. Results of the Performance Evaluation

The following chapter describes an analysis of the convergence of the street segment representation. All segments have been derived as result of the merge from several journeys for the described application scenarios.

A. *Motorway*

The two highway sections shown in Figure 9 are the A43 and the A40 crossing in the bottom right at Bochum junction (GPS-coordinates approximately: 51.516 N, 7.206 E). Figure 9 depicts a map and a corresponding Google Maps screenshot in an increased diameter. Recorded average speeds below 80 km/h are indicated by a dashed, faster segments by a solid line type. The grey circles mark deduced crossings between at least two map segments. The gap in

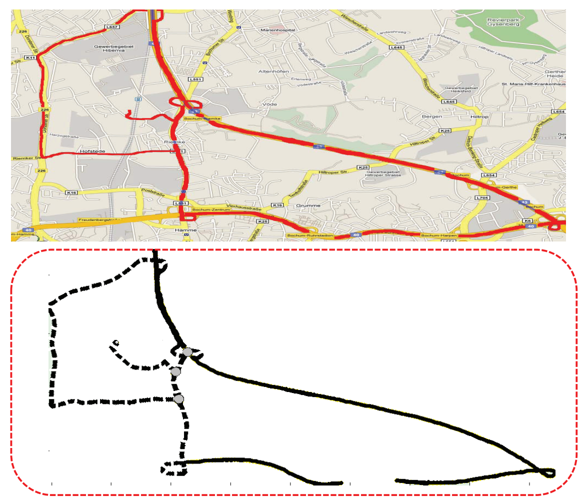

Figure 9. Reference check of deducted map

the A40 segment results from a tunnel between the exits Bochum-Harpen and Bochum-Ruhrstadion. To clarify the original route and the reached accuracy, Figure 9 (top) shows a Google Maps screenshot in an overlay with the driven route.

This vivid example clarifies the achieved approvements. In contradistinction to the Google Maps data origin, the map generator must not work with specially equipped time- and cost-intensive measurement cars. Concerning the fact, that our map generator just uses a high quantity of CND position tracks (e.g., from mobile phones), we achieve a very prompt overall map, demonstrating an accuracy which is competitive to that of Google.

Figure 10 depicts the average error of a map segment related to the underlying journey count. At this point, the authors want to mention, that the used journeys were tracked with an external GPS antenna, supporting Differential GPS (DGPS). As reference we used the entire street width. Hence, a deviation ΔD of 0.20m does not correspond with an accurate localization (compare with Figure 10), it just figures out how accurately and how fast the map generator is able to create a realistic street position and -course. Based on

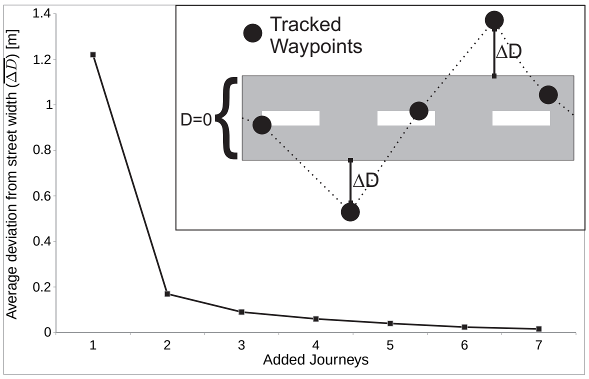

Figure 10. Mean deviation from roadside vs. journey count

this error measurement, we assume that merging 4 to 6 segments describing the same street can result in a sufficient representation ($\overline{\Delta \text{D}} \leq 0.1$m) of a street segment existing in reality.

B. City Traffic

Figure 11 shows the evolution of the map generation process for the evaluated city scenario. It can be seen that the resulting map perfectly matches to the Google Maps view. The evolutionary map generation process is depicted in three major steps, following additional journeys taken into account for the computation.

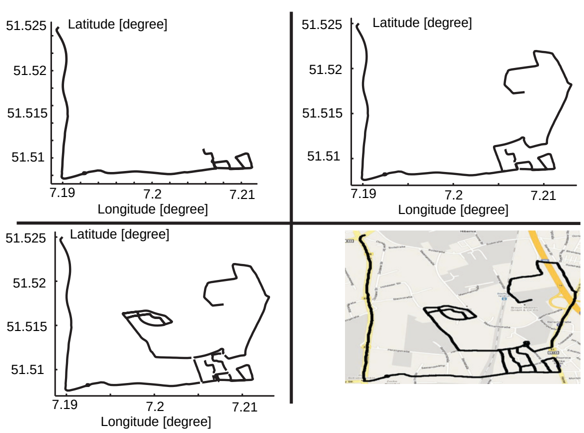

Figure 11. Map Generation for the City Scenario

For a productive service the effect of convergence and thus the effort up to an initial map generation is important. Claiming that a map is ready for use when new data has not caused major changes in the database for some journey additions, we focus on the quantity of information particular journeys contribute, named $I(J)$.Therefore, we analyse after every merge process, how many new nodes have been contributed by the actually beholded journey to the overall map. In case of a complete new segment, every node carries information about a street-part, which was not detected yet ($I = 1$). On the other hand, it is also possible, that some journeys describing already detected parts of the road network, so the information content of such journeys is $0 \leq I(J) < 1$.

In general these results will depend on many factors such as the quality of available journey data, too. However, this test of the processing gives a good insight into the common map generation.

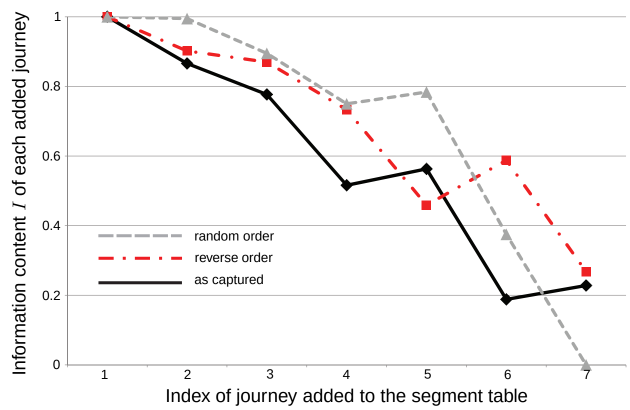

Figure 12. Information content $I(J)$ of each journey added in three different orders

Figure 12 depicts the results for three different processing orders of 7 journeys on the paths shown in Figure 11. The area covered is of about 0.5km^2. It can be seen that the information content of a new journey is depending on the number of already analyzed journeys. Beneath the order, other parameters like position and length of journey, do also have a high influence on the information content. A short journey on a main road do not have a hugh information content to an already created segment table, whereas a long journey, which just uses small side roads, can depict a very hugh information content, even when merging such a journey to an already existing map.

These effects are also visible in Figure 12. In every order the first added journey shows an information content of $I = 1$ and the fifth journey of the random order, for example, has a higher amount of information content than the forth one, though it was added afterwards.

It has to be noted that this test is only a first hint and that the authors are aware of the fact, that the absolute amount of information in a journey will show significant variations depending on a multitude of parameters such as length of journey, precision of data, novelty of taken path etc..

C. Pedestrian Movement

In order to demonstrate the accuracy, scalability and behavior of the map generation framework for lower speeds and therefore a geographically dense set of waypoints, Figure 13 depicts the result of a journey collection and map generation process on our campus. The journeys have been

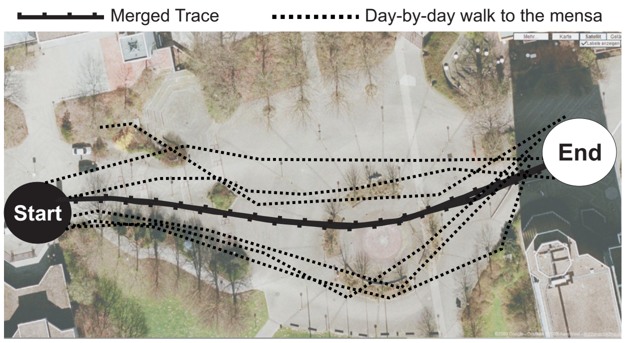

Figure 13. Merged Segment in comparison to measurements

collected by a pedestrian user walking different ways to the refectory. If a map for this scenario is generated, the merged journeys lead to a segment located in the middle of the journeys, as expected. It is also visible that outliers in the measurement or movement pattern are mitigated by the number of journeys taken into account. Again, the resulting map is depending on the accuracy of position estimation, which obviously is high in this scenario, as different movement paths can be detected clearly. This benefit is gained by the combined localization process and the sophisticated GNSS receiver of the iPhone.

D. Indoor Localization Testbed

This experiment again give an insight into the performance of the filtering and merging algorithms as being described in Section III. Figure 14 depicts the preliminary results of the journey collection without applying the algorithms of the map generator. The position deviation from the train lane is significant and obvious. Furthermore, it is fluctuating for each circular driving under static conditions. We have interconnected three independent cycles, which explains the starting and ending points of the journeys. As depicted in the figure, the RF-based indoor localization system shows position dependent fluctuations in the accuracy of position estimation. The overall performance of the system shows a maximum error of around 1m, where in most cases the accuracy is considerably better.

Taking into account journey 3 from Figure 14, Figure 15 shows the result of the consistency checks alone before the merging to the final map is executed. It can be seen that outlier measurements are mitigated. The consistency checks have been performed using the following parameters:

1) Speed Limit: $1.5\frac{m}{s}$
2) Acceleration Limit: $0.5\frac{m}{s^2}$
3) Vector Plausibility: $40°$

In the next step, we executed the intra journey merge routine in order to generate a more accurate map of the

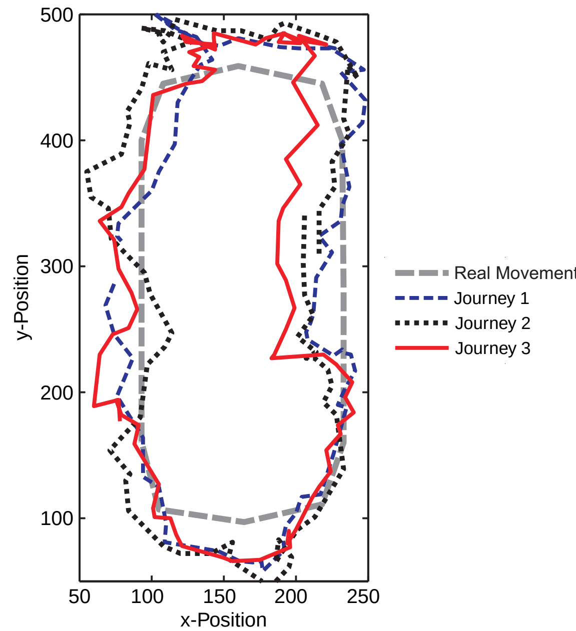

Figure 14. Comparison of measured journeys (raw-material) and the real movement

scenario. In this case we merged three journeys after the plausibility checks. In Figure 16 it can be seen that the map has closely adapted to the real train lane. It is also visible that constant fluctuations of the RF localization affect the accuracy of the map. As a plus, these adverse effects seem to be confined to the area of their incident. In comparison to the raw data, the generated map greatly enhances the accuracy of the localization process. Summarizing, the accuracy of the map reduces the maximum error by 0.25m in this case.

In the final system design, the map generation algorithm can be used to generate an up to date map of the application scenario. When thinking of an industrial or public indoor environment, the available maps are as old as the building itself. Constructional changes are often not reflected in building plans. Hence, these blueprints are not usable for optimizing the performance of an indoor localization system.

After generating the map, the user position can be estimated by a combination of CM and localization system by applying a particle filter approach [11]. The positions estimated by the system are represented in so called particles, which are weighted and mapped on the new building map. By doing so, outliers can be mitigated during the operational phase of the system, but hence the degree of freedom is reduced. Concluding, a certain update interval for the user map is mandatory to retain the accuracy of localization.

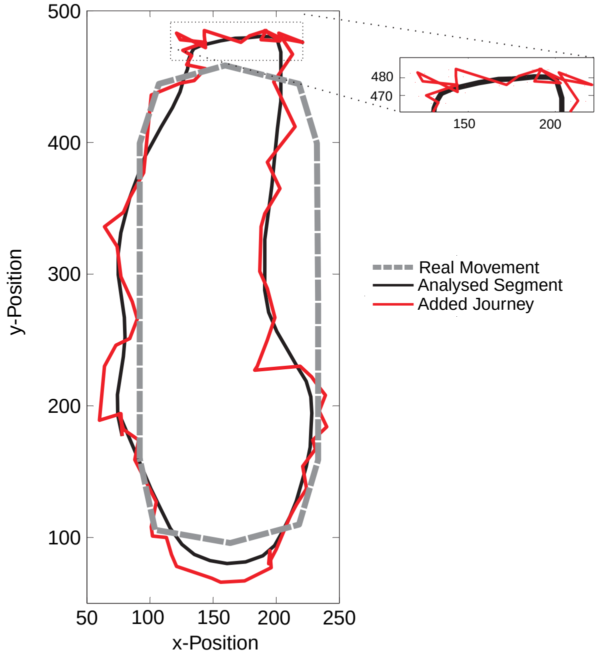

Figure 15. Raw Material before and after consistency checks

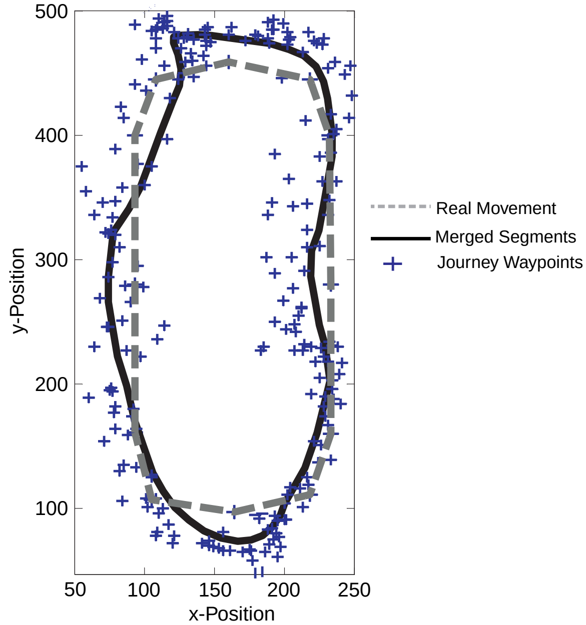

Figure 16. Merged Segment in comparison to measurements

VI. Conclusions

This paper describes a successful experimental implementation of map generation out of GPS journeys recorded by several mobile subscribers. Employing attribute tables for each street segment should enable an enhanced navigation service. The combination of the Java Applet and the Generator Server Program allows the user to create map data plus lots of other useful information for navigation. The initial results of these experiments suggest that only 4 to 6 journeys along a particular street segment result in a useful estimation even in complicated scenarios. Furthermore, we have demonstrated the scalability of the proposed map generation framework by exploring four scenario classes. The results demonstrate the wide applicability of the approach. Especially indoor scenarios can benefit from this framework. The mean position deviation has been reduced considerable.

These promising findings motivate further extensions, including 3D-Navigation by means of additional sensor data (like accelerometers) or optimizations for noise reduction and increased accuracy. Power awareness for long battery lifetimes and communication protocol dependent topics like traffic models or business case schemes will be studies in upcoming work. Additional features as described in [6] will enhance the controllability of the consumer hardware. The influence of the sample interval to the fidelity of the resulting map has to be carefully pondered against the resulting workloads (in terms of bandwidth usage, storage requirements etc.) in the system and on the radio link.

VII. Future Work

The map generation framework is capable for opening new application fields for traffic management and indoor localization applications.

The next step will be that the map generator algorithms are optimized for real-time calculation. If this target is reached, today's traffic forecast procedures can even benefit, if e.g., working sites on motorways can be detected out of the transmitted journeys. This is a realistic target, as the GNSS accuracy will be enhanced with the launch auf Galileo. Furthermore, if a temporal connection between navigation devices and traffic management center can be established e.g., by using public cellular or satellite networks, journeys and route planning can be transmitted on the uplink and be regarded for traffic jam prognosis; updated traffic information for adaptive routing and lane assistance can therefore be forwarded to the navigation devices of the user as a special service of the operator.

The applied consistency checks are very useful for indoor localization systems. If the calculation time is optimized, these algorithms can be applied in a local control center. As we have seen that the accuracy can be enhanced, this approach will lead to reduced costs, as the investment in sophisticated localization devices can be reduced.

Acknowledgment

This work is conducted within the SAVE Project (Geographic Information System for Alarming using autonomous, networked Gas Sensors) and is funded by the German Federal Ministry of Education and Research (BMBF) – 16SV3711

REFERENCES

[1] Niehöfer, B., Burda, R., Wietfeld, C., Bauer, F., and Lüert, O., GPS Community Map Generation for Enhanced Routing Methods based on Trace-Collection by Mobile Phones, Advances in Satellite and Space Communications, 2009 (SPACOMM), pages 156-161, Colmar, France, 2009

[2] Bruntrup, R. and Edelkamp, S., Incremental Map Generation with GPS Traces, Intelligent Transportation System, 2005. Proceedings. IEEE, pages 574-579, 2005

[3] Edelkamp, S. and Schroedl, S., Route Planning and Map Inference with Global Positioning Traces, Computer Science in Perspective, pages 128-151, Springer Verlag, 2003

[4] Kaasinen, E., User needs a Location-Aware Mobile Services, Personal and Ubiquitous Computing, Vol. 7, pages 70-79, Springer Verlag, 2003

[5] Duckham, M. and Kulik, L., Simplest Paths: Automated Route Selection for Navigation, Lecture Notes in Computer Science, Vol. 2825, pages 169-185, Springer Verlag, 2003

[6] Robbins, D. , Cutrell, E., Sarin, R., and Horvitz, E., ZoneZoom: Map Navigation for Smartphones with Recursive View Segmentation, AVI '04: Proceedings of the working conference on Advanced visual interfaces, pages 231-234, Gallipoli, Italy, 2004

[7] Schroedl, S., Rogers, S., and Wilson, C., Map Refinement from GPS Traces, Technical Report RTC 6/2000, DaimlerChrysler Research and Technology North America, 2000

[8] Zhao, Y., Standardization of Mobile Phone Positioning for 3G Systems, Communications Magazine, vol. 40, issue7, pages 108-116, IEEE, 2002

[9] IEEE 802.15.4 Workgroup, IEEE Standard for Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), New York, NY, IEEE, 2003

[10] IEEE 802.15 Workgroup, IEEE 802.15.4a Standard, WPAN Low Rate Alternative PHY Task Group 4a: IEEE Standard for Wireless Personal Area Networks Low Rate Alternative Physical Layer (PHY), 2007

[11] Widyawan, M.K., Klepal, M., and Pesch, D., A Bayesian Approach for RF-Based Indoor Localisation, Wireless Communication Systems, 2007. ISWCS 2007. 4th International Symposium on, pages 133-137, Trondheim, Norway, 2007

[12] Kang, H., Seo, G.w., and Lee, J., Error Compensation for CSS-based Localization System, Proceedings of the World Congress on Engineering and Computer Science 2009 Vol II WCECS 2009, San Francisco, USA, 2009

[13] Enderle, W., Weber, T., and Kuhlen, H., The Architecture of the European Global Navigation Satellite System - GALILEO, 10th Australian International Aerospace Congress incorporating the 14th National Space Engineering Symposium, Brisbane, Australia, 2003