Challenges and Opportunities of EGNOS/EDAS Augmentation and Integrity Computation in the Cloud

A. Favenza, Istituto Superiore Mario Boella
M. Pasin, Microsoft Innovation Center Torino, Italy
G. Marucco, J. Huang, Istituto Superiore Mario Boella
F. Scullino, Microsoft Innovation Center Torino, Italy

BIOGRAPHIES

Alfredo Favenza is a senior researcher of Mobile Solutions Area of ISMB and is involved in research activities in the areas of navigation technologies, GNSS Software Defined Radio, Galileo and EGNOS/EDAS in collaboration with NavSAS group. He is active in several R&D projects, where cloud computing, big data, mobile devices and navigation technologies are combined to generate innovative value-added solutions.

Marco Pasin is the head of Microsoft Innovation Center Torino, which he joined in 2010 as senior SW architect and developer. He previously worked as a as a researcher in the Navigation Technologies team (NavSAS) of ISMB. His skills focus primarily on cloud computing architectures and applications, mobile applications and embedded systems. He is also a teacher on Microsoft Azure for Microsoft Italia.

Gianluca Marucco is responsible of the GNSS Platforms for Safety and Environment research unit of ISMB Navigation Technologies team (NavSAS). His current research interests are focused on different aspects (including Safety) of the GNSS augmentation systems: both Satellite and Ground based like the aforementioned EGNOS and the differential regional networks.

Jie Huang is a researcher in ISMB, Turin, Italy. He did his PhD degree in the department of electronics and telecommunication of Politecnico di Torino from 2011 to 2013. His PhD activities focused on developing DVB-T positioning receivers and integration of GPS and DVB-T, cyclic slip detection on dual frequency receiver. Now he is working on precise positioning based on GNSS and EDAS.

Francesco Scullino is a researcher of ISMB since 2010. He has been involved in mobile application development for research activities in the health field. Since 2012 he has been working at Microsoft Innovation Center of Torino focusing his activity on web application development in cloud computing environment. Nowadays he is also a member of the BizSpark team that provide technical support concerning Microsoft Azure technology.

ABSTRACT

Highly accurate positioning is a crucial element for the development of various categories of mass-market and professional position-based applications such as some Location Based Services (LBS), Road Applications, High Precision Agriculture and Disaster Alerting. By 2022, according to the GNSS European Authority (GSA) Market Report (October 2013) [1], LBS will represent the largest market in GNSS (47%) followed by Road segment (46.2%). Moreover, thanks to a continuing reduction in costs and increase in performances of satellite navigation receivers, the dependence of these applications on satellite navigation systems is expected to grow further, especially in the mass-market segment, which is driven by the explosion of mobile and embedded devices like smartphones and tablets.

In this scenario, the EGNOS/EDAS system represents a key technology to improve the positioning performances of GNSS mass-market receivers. The EGNOS/EDAS system can increase the accuracy of position measurements by transmitting information that correct satellite navigation data and by providing an evidence of positioning reliability through the integrity messages.

For all these reasons, engineers and researchers in the field of GNSS are exploring alternative methodologies to exploit the benefits deriving from the use of EGNOS/EDAS. New opportunities are arising, among others, from the integration of the GNSS technologies with emerging ICT technologies, e.g. cloud computing.

In this paper, the authors want to propose and analyse a cloud-based architecture for centralized computation of augmentation and integrity based on EGNOS/EDAS information services. This solution aims to demonstrate that this innovative approach could represent a valid alternative to the standard access to EGNOS through Signal-In-Space (SIS), providing a centralized platform which is able to serve a large number of devices, via standard terrestrial communication channels, with high-accuracy positioning based on reliable EGNOS/EDAS services in a Navigation-as-a-Service (NaaS) approach.

INTRODUCTION

Exploiting the capabilities of EGNOS/EDAS may often be not straightforward and can represent a challenging task for several important reasons. Firstly, the reception of EGNOS SIS can be problematic because of different causes, in particular adverse environment, as in urban scenarios, where the EGNOS satellites elevation is relatively low and the reception of the signal is rather limited. Secondly, for the exploitation of EGNOS capabilities through the EDAS system [2], a service provider which makes available EDAS information to final users is needed. Furthermore, the user device must be capable to accept and elaborate such information, in order to calculate augmented position including integrity information. Finally, due to the complexity behind the calculation and application of EGNOS corrections, sub-optimal implementations of the algorithms are likely to occur on end user devices, leading to erroneous PVT and protection levels computation, in particular when applications are exempted by any sort of certification/approval.

Cloud computing offers the advantage of having a high performance and distributed architecture through a number of connected computing units. In such environment software, platforms and infrastructures can be delivered on-demand to users through the Internet.

The use of cloud computing allows to reach high levels of reliability, flexibility and scalability to provide stable and highly-available augmentation and integrity services, mitigating the fallacies deriving from the use of the EGNOS described above and avoiding the threats deriving from bad signal reception. This Navigation-as-a-Service (NaaS) approach enables the delivery of new navigation services based on cloud infrastructure following the paradigm of Value-Added Reseller (VAR) model.

In the following sections, the authors will describe the architecture of the proposed solution and investigate the potential performance challenges introduced by the adoption of a distributed computing architecture, as opposed to the opportunities that are brought in the exploitation of the EGNOS/EDAS capabilities, through a comparison of the overall performances of a standard device-based solution versus the proposed cloud-based solution in terms of accuracy, availability, and integrity of positioning. The applicability of this study can find a significant scenario in the context of vehicular networking, where a terrestrial communication channel is supposed to be always available and with a particular consideration for Northern European countries where the use of EGNOS through SIS is limited due to low satellites visibility. Most part of this study has been conducted in the framework of the FLOODIS project [3], a European project partially funded by the European Commission, focused on the integration of GMES (Global Monitoring for Environment and Security) emergency services with satellite navigation and communication for establishing a flood information system. Finally, even if this paper is focused on Wide Area Augmentation Systems (WAAS), the presented model could be also easily adapted to build services based on Local Area Augmentation Systems (LAAS).

EGNOS/EDAS SERVICES

As shown in Figure 1, the EDAS service provides the augmentation data through a variety of services [2]:

- EDAS SISNeT service, which provides access to the EGNOS GEO satellites messages transmitted over the Internet through the SISNeT protocol.
- EDAS Data Filtering Service, which allows EDAS users to access a subset of the Service Level 0 or Service Level 2 data (data available in ASN.1 and RTCM 3.1 formats respectively).
- EDAS FTP Service, which enables EDAS users to get EDAS/EGNOS historical data in different formats and data rates.
- EDAS Ntrip Service, which provides GNSS data (RTCM format) coming from the EGNOS network through the Ntrip protocol in real-time.

In this paper we will focus in particular on these services:

- EDAS Ntrip Service [4], to get real-time EDAS information to be broadcasted to embedded devices. Ntrip (Networked Transport of RTCM via Internet Protocol) protocol is an RTCM standard (RTCM 10410.0) [5] designed for disseminating differential correction or other kind of GNSS streaming data to stationary or mobile users over the Internet. Ntrip is becoming the "de facto" protocol for GNSS data dissemination in real time. It supports wireless Internet access through Mobile IP Networks like GSM, GPRS, EDGE, or UMTS. As a result, data are available without requiring direct access to an EGNOS satellite, which is very useful in constrained environments such as where signals are blocked or are disturbed by interference.
- EDAS FTP Service [6], to get historical EDAS information and enable cloud computing post-processing.

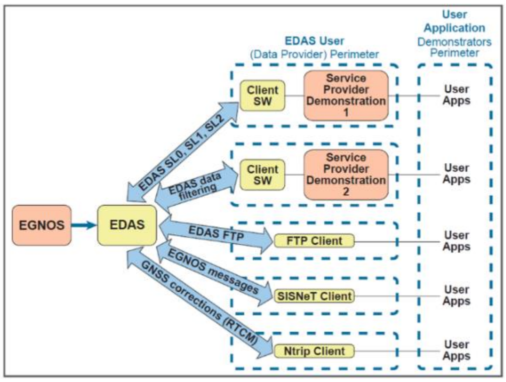


Figure 1. EDAS Architecture and Data Stream Services

CLOUD COMPUTING AND PLATFORM AS A SERVICE

Cloud Computing is a computational model which enables on-demand access to a shared resource pool through the network. These highly configurable resources can be easily accessed and released with a minimum management effort and a limited interaction with the service provider [7]. Cloud computing solutions come with characteristics of high availability, scalability and performances representing a very profitable opportunity for the industrial sector and allowing to reduce effort, time and costs of development, distribution and management.
Cloud computing solutions are usually classified in three service models:

- Infrastructure as a Service (IaaS): the service provider delivers the complete framework of servers, routers, storage, hardware and virtualization software; the user is responsible of operating system, middleware, runtime and applications.
- Platform as a Service (PaaS): the service provider delivers the whole hardware and software chain which includes networking and runtime functionalities; the user is responsible for data and application management.
- Software as a Service (SaaS): the service provider delivers the whole service, including applications and data; the user just uses of service's functionalities.

The Microsoft Azure [8] platform has been selected as cloud computing service provider for the FLOODIS project and therefore also for this scientific study. This choice is comes from the fact that the Microsoft Azure platform offers native support to the PaaS model, which consents to reduce setup and management costs and to accelerate solution's time-to-market. Moreover, Microsoft Azure provides a wide range of development technologies, which results in a higher flexibility in project design and development phases.
Microsoft Azure platform provides a set of cloud computing services which can be used separately or combined together in order to build remote applications based on complex architectures. Main services offered by the platform are remote machine virtualization, a storage system based on non-relational tables, files and queue messaging, SQL Azure relational databases and a set of extra functionalities like an access control service and a remote distributed caching system.

EMBEDDED CUSTOM DEVICE

As described in the introduction, the objective of our analysis is to highlight the possible advantages coming from moving the EGNOS/EDAS augmentation computation from user device to a cloud-computing service. Available mass-market personal devices (i.e. smartphones and tablets) could represent very useful platforms to test our approach on extensive use. Unfortunately, the GNSS chipsets integrated into these devices provided information just limited to a subset of NMEA messages, without access to low level raw navigation data, such as pseudoranges, Doppler Effect and carrier phase estimations. For this reason, our choice was to assemble an embedded custom device (see Figure 2) based on an ARM® Cortex™-A8 core board connected to a GNSS receiver able to provide raw data in a proprietary format as shown in table.

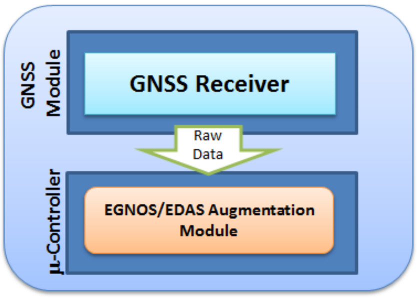


Figure 2. Diagram of embedded custom device configuration

In this scenario, the board can collect raw data from the receiver and use them to calculate augmentation in two different operational mode:

- "*on-board*", directly on the embedded device collecting EGNOS information from SIS
- "*in-remote*", transmitting raw data from the GNSS receiver to the cloud platform: here an augmentation module is continuously receiving correction messages from EGNOS via EDAS services and collecting them in remote storage, ready to be used "on-demand" to perform a remote augmentation computation.

In the second operational mode, at the end of augmentation computation, PVT results and vertical and horizontal Protection Levels are returned to the custom device via terrestrial communication.
The picture in Figure 3 shows the connected hardware components of our embedded custom device.

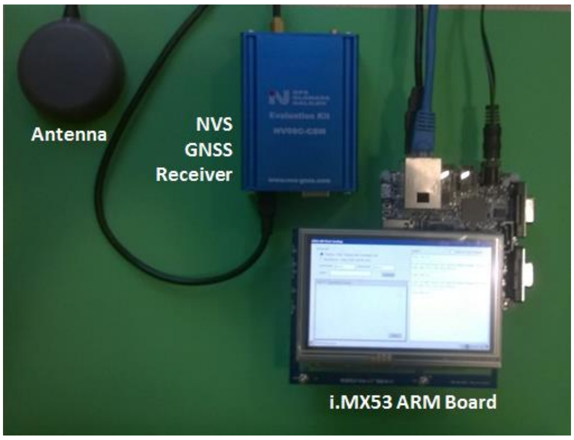


Figure 3. Embedded System Components

The ARM board used for the testing consists in a Freescale i.MX53 family, a power-efficient implementation of the ARM® Cortex™-A8 core, optimized for both performance and power to meet the demands of high-end, advanced applications, ideal for a broad range of applications in the automotive and industrial sectors. The ARM board is linked, via USB connection, to the GNSS receiver, the EVK MCM evaluation kit from NVS, embedding a mass-market multi-constellation 32 channels receiver able to output raw data in a proprietary format which could be easily converted in standard RINEX by using the RTKlib [9] software suite. The receiver is connected to an external antenna and is able to process the EGNOS SIS, decode the integrity message and output Horizontal Protection Level (HPL) and Vertical Protection Level (VPL) following the standard algorithm reported in Appendix J of RTCA [10].

AUGMENTATION ALGORITHM DESIGN

The implemented software module exploits capabilities of EGNOS through the reception of EDAS data.
The core algorithm for the augmented positioning is essentially based on the standard definition and description [10]; what is not included in the MOPS is the part concerning the retrieval of the augmentation information that is done through EDAS [11]. The algorithm includes a short time data storage in order to be able to select the correct piece of information when needed; such selection is done on the basis of the measurements done on the GNSS satellites SIS and provided by the receiver in real time or through EDAS. Once valid information is retrieved, the components of the correction and the residual error estimate can be computed. These values combined with data coming from the receiver allow to compute the augmented position and the related Protection Level.

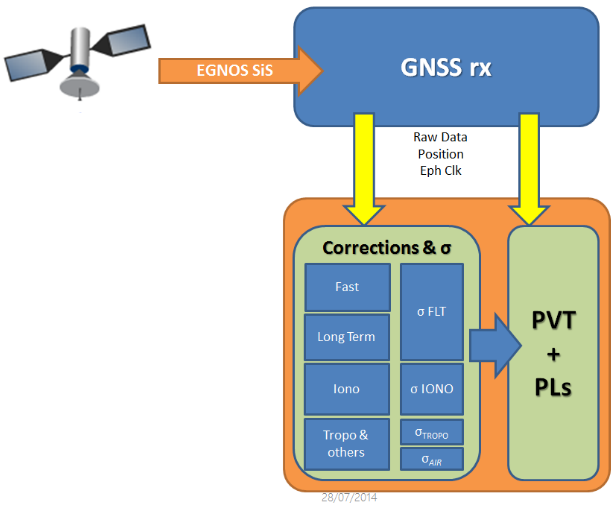


Figure 4. Mid-level block diagram describing augmentation algorithm structure and data flow

PROPOSED CLOUD-BASED APPROACH: THE AUGMENTATION MODULE

The core of the proposed cloud architecture is a software module, the Augmentation Module, deployed on a Microsoft Azure cloud instance, which is responsible of two main tasks:

- Collect RTCM messages received via the EDAS client [12] and store them on a remote database; this is a process which runs continuously in order to guarantee the presence of fresh EGNOS correction when an incoming positioning request from the embedded device has to be managed;
- Execute – on demand – our implementation of EGNOS augmentation algorithm using GNSS raw data from the embedded device and appropriate EDAS correction extracted from database as input; this task generates in output a highly accurate PVT and a protection level value which can assure the reliability of device positioning.

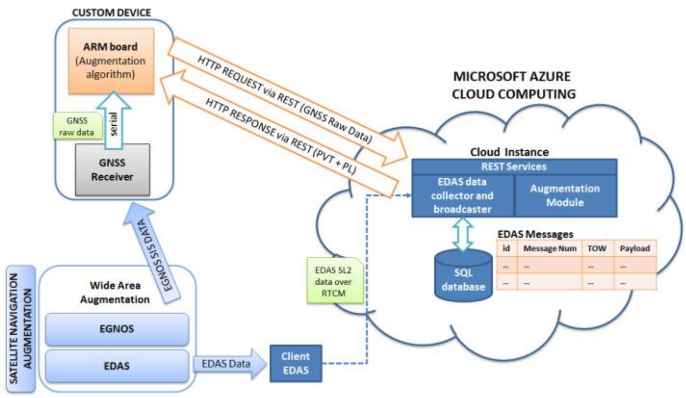


Figure 5. Architecture of the cloud platform and connection with embedded device and EGNOS/EDAS services

In order to perform these tasks, two HTTP REST services have been implemented to allow the embedded device and the cloud platform to exchange information needed for PVT and protection level computation. The activation of these services relies on the transmission of REST requests and responses. REST (REpresentional State Transfer) [13] is an architectural style for resource exchange born around 2000 which focuses on resources description, their identification in the web and the transmission through the nodes of the network.
On device side, the i.MX53 board collects raw data coming from the NVS receiver and sends a REST request to the cloud service via HTTP. Table 1 provides an example of a REST request message, consisting in a device ID and three arrays for raw measurements, ephemeris and Ionospheric coefficients extracted from the receiver.

Table 1. Example of REST message for positioning service request to cloud platform

"DeviceID":	1
"measures"	{"measuredata":"2.1993200000e+05,1793,On,Off,On,Off,,G04,10,GPS_L1CA,Main,437,Off,Off,Off,4.0250000000e+01,2.4956074798e+07,2.5656394000e+03,1.3114506833e+08"}
	{"measuredata":"2.1674200000e+05,1793,On,On,On,Off,,G12,3,GPS_L1CA,Main,4485,On,On,Off,4.6750000000e+01,2.2034420065e+07,2.8199910000e+03,1.1579175992e+08"}
	… (a row for each satellite in view)
"ephemerisClocks":	{"ephclockdata":"2.1674200000e+05,1793,4,769,1,0,0,0,19,19,19,0,-6.5192580223e-09,223200,0.0000000000e+00,1.1368683772e-13,6.9183297455e-06,-1.8187500000e+01,1.5556906874e-09,-1.3424507156e-01,-7.6368451118e-07,1.0554343578e-02,1.0374933481e-05,5.1537360630e+03,223200,-1.6763806343e-08,-5.0404657610e-01,1.5273690224e-07,2.9860382341e-01,1.7034375000e+02,3.4938043496e-01,-2.6577708923e-09,1.0334133549e-10,769,769"},
	{"ephclockdata":"2.1674200000e+05,1793,12,769,1,0,0,0,58,58,58,0,-1.2107193470e-08,223200,0.0000000000e+00,2.5011104299e-12,2.0615849644e-04,1.2968750000e+01,1.3119461073e-09,2.5750044733e-01,7.4133276939e-07,5.1384844119e-03,3.9916485548e-06,5.1536221924e+03,223200,1.1175870895e-08,8.4464268247e-01,1.1920928955e-07,3.1437536888e-01,3.1462500000e+02,1.3783091260e-01,-2.5834197004e-09,-1.2164491636e-11,769,769"},
	… (a row for each satellite in view)
"ionos":	{"ionodata":"2.1674200000e+05,1793,26,1.4901161194e-08,2.2351741791e-08,-1.1920928955e-07,-1.1920928955e-07,1.0854400000e+05,1.3107200000e+05,-1.3107200000e+05,-3.9321600000e+05"}

On cloud side, the Augmentation Module running on the cloud service takes data from the device REST request and inputs them, together with pre-stored EDAS data, to the augmentation computation procedure. At the end of this task, position, HPL and VPL are calculated and transmitted to the embedded device via the REST response described in the example in Table 2.

Table 2. Example of REST message for positioning service response to embedded device

"lat":	"4472420.8024"
"lon":	"601433.9839"
"height":	"4492695.7516"
"hpl":	"1.9"
"vpl":	"1.7"

At this stage, the client-server communication is closed. By As shown in Figure 6, the GNSS receiver connected to the board generates a new set of raw data request each second. In order to provide a continuous positioning service and test the responsiveness of this cloud-based approach, the embedded device has been configured to send a REST request at each second.

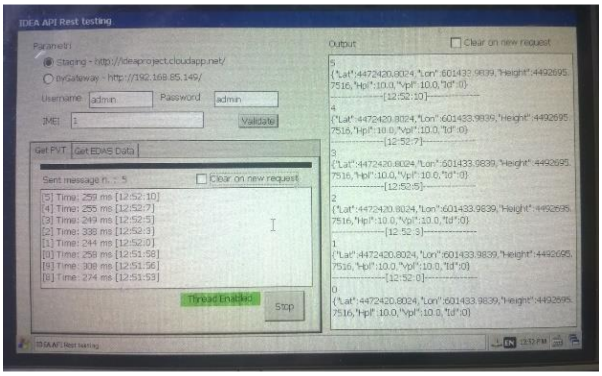


Figure 6. Screenshot of device UI showing positioning responses obtained by the custom device at each second

In the next section, some results of experimental tests are presented, where the performances of the proposed solution have been assessed on the basis of a set of pre-defined KPIs.

EXPERIMENTAL TESTS

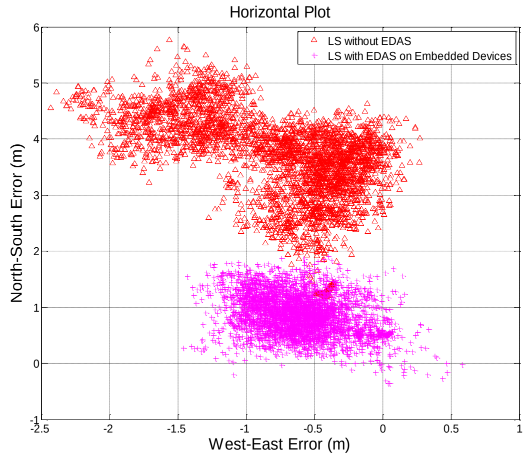
In order to evaluate performances of proposed cloud-based augmentation solution, several Key Performance Indicators (KPIs) have been introduced:

- *Precision*, *accuracy* and *reliability*: this indicators help to assess positioning accuracy and protection level performances in both tested solutions;
- *Availability*: this indicator is used to compare the time requested to have all needed EGNOS information available when the device requests the positioning service. This comparison is made between the cloud based EDAS approach and the device based EGNOS SIS approach;
- *Responsiveness*: this indicator is used to test the rapidity of the proposed cloud-based solution to respond to an embedded device request with PVT+PL data and to detect performances degradation. This value can be affected by three elements: execution time of Augmentation algorithm on the cloud, data transmission delay between an embedded device and the cloud due to networking communication latency and data retrieving of EDAS messages from the remote cloud database.

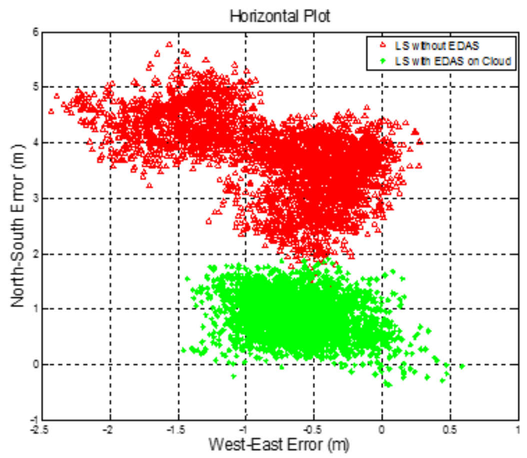
All tests have been conducted in post-processing with a 1 hour long set of data collected from the antenna attached to the NVS receiver connected to our Freescale i.MX53 embedded board.

Precision, accuracy and reliability test

The augmentation algorithm has been tested against a real GPS and EDAS dataset. The GPS data has been collected through a fixed antenna on the roof of ISMB. In this test, only GPS satellites have been used and the EDAS data are received directly from the EDAS FTP server, both on the embedded device and in the cloud service. As expected, on the side of position, precision and accuracy, same results in both cloud and device scenarios have been found.



(a)

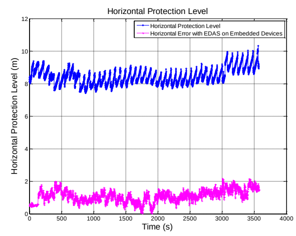


(b)

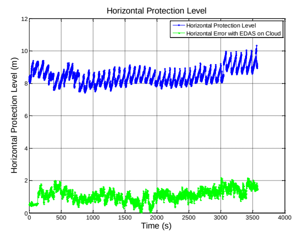
Figure 7. Positions estimated by Least Square method with and without EDAS in embedded device (a) and in cloud computing (b)

In Figure 7, the two dimension plots are shown. Here the Least Square method is used for the PVT solution. It is clear that the bias of estimated positions decreases when EGNOS/EDAS correction is applied and also the estimated positions converge to a smaller area.

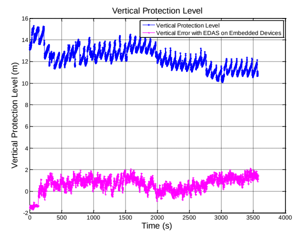
In Figure 8, we plot the horizontal and vertical protection level for both device-based, (a) and (c), and cloud-based, (b) and (d), computational approaches. In the protection level calculation, σ_FLT and σ_IONO are derived from EGNOS/EDAS message. It is clear that the error is always correctly bounded by the protection level and the result is the same for both solutions.



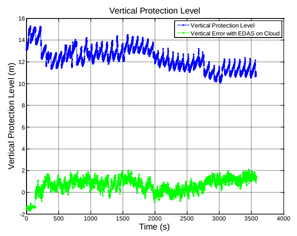
(a)



(b)



(c)



(d)

Figure 8. Horizontal and Vertical Protection Level

Availability test

In these tests we want to evidence the benefits coming from a cloud-based approach in terms of availability of service enhancement. The implemented cloud platform, with respect to the direct access to EGNOS SIS from the embedded device, relies on a cloud database system where RTCM messages collected via EDAS services are stored also when the system is idle: this approach allows users to have, in every moment, all needed augmentation data available on-demand. This represents a relevant advantage with respect to the direct access to EGNOS SIS: in fact the availability of EGNOS SIS, especially in northern countries, is low because of a poor EGNOS signal reception; a system which can guarantee the exploitation of EDAS via cloud computing could significantly enhance the robustness of positioning. Moreover, even in areas where EGNOS satellites visibility is good, the time to get the whole set of data corrections could be very long, more than 2 minutes in the worst case [10]. From the tests on our system in full operation (i.e. after the initial setup of the database), we experienced a considerable reduction or elimination of this waiting time, ensuring a high availability of augmentation service provisioning.

Responsiveness test

The aim of this test is to analyse the performances of our augmentation algorithm when executed in a cloud computing environment. In order to obtain a positioning service which is responsive in terms of time, we want to meet the target of 1 second to process and output each PVT+PL result set, resulting in the provisioning of a position information per second. This target time includes the computation interval, the time needed for data networking and the retrieval of EDAS corrections stored in the cloud database.

Responsiveness test: algorithm execution time performances

The cloud computing tests have been performed on a PaaS architecture on different service configurations, which are characterized by different system specifications. Table 3 summarize the list of tests with the chosen setups and related average execution times:

Table 3. Summary of algorithm execution performances tests conducted with different cloud service configurations

Test Id.	Instance type	CPU freq. (MHz)	Num. of cores	Mem (MB)	Avg. exec. time (ms)
T1	Extra Small	1000	1 **(shared)**	768	1131
T2	Small	1600	1	1750	502
T3	Medium	1600	2	3500	492
T4	Large	1600	4	7000	499
T5	Extra Large	1600	8	14000	494

The first test (T1) has been conducted deploying our solution on an Extra Small Microsoft Azure cloud instance, which is the one with the lowest performances. With this configuration, the cloud solution has, as expected, very poor performances: PVT + PL computation time fluctuates between 1000 and 1400 milliseconds of elaboration (see Figure 9a), which is not acceptable with respect to our 1 second target. Is important to remark that, with this type of service, the CPU is not dedicated, but shared with other compute instances running in the cloud datacenter. In the second test (T2), we tested a better performing cloud configuration deploying the solution on the next available compute tier, i.e. Small instance with one dedicated CPU. T2 test results evidence that, with this type of configuration, execution time decreases significantly: a deep analysis of the profiling of the algorithm via the Visual Studio Diagnostic Tools [14] shows that the execution of a PVT with Least Squares and the EDAS augmentation and integrity computation remains, in the T2 testing environment, around a computation time of ~500 millisecond (see Figure 9b), which can be considered acceptable to meet our target of 1 second for the delivery of complete positioning computation and transmission chain.

In the third test (T3), we tried to further enhance the cloud configuration deploying the solution on Medium cloud instances configured as in Table 4. At shown in Figure 9c, at this stage no further relevant performance improvements can be noticed: our conclusion is that the computational limit due to algorithm complexity and its software implementation has been reached and we cannot expect significant improvement augmenting the performances of the cloud service. This is confirmed in tests T4 and T5, conducted with high performances cloud instances, which do not shown any substantial improvement in average execution times (Figure 9d and Figure 9e).

Figure 10 shows in a comparative way the results of the previously described tests, evidencing the presence of a computational threshold which limits the possibility to enhance performances over a given limit simply increasing the computational resources.

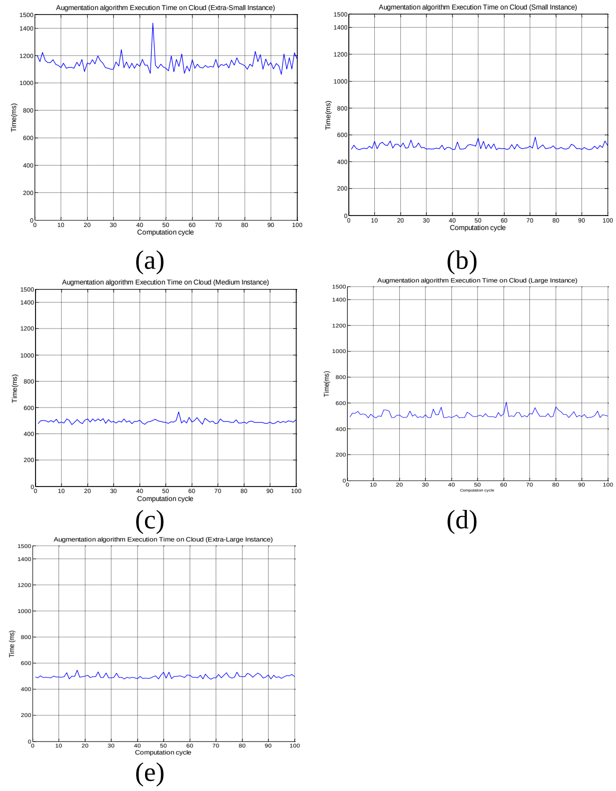


Figure 9. Augmentation algorithm execution time with Extra-Small (a), Small (b), Medium (c), Large (d) and Extra-Large (e) cloud instance

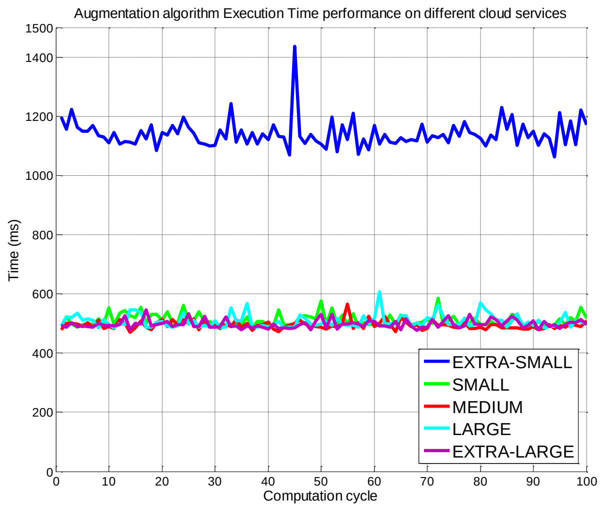


Figure 10. Execution time performances comparison with different cloud service configurations

The cloud computing environment performances results have then been compared to the performances of the algorithm executed on our embedded custom device, in order to confront the effectiveness of our solution in terms of computing resources requirements. The embedded device used for this test is based on an ARM® Cortex™ A8 CPU with this configuration:

Table 4. Embedded device hardware configuration

Core Operating Frequency (MHz)	Number of cores	Memory (MB)
800	1	512

The custom embedded system is able to compute positioning and augmentation within our target time, taking about 550 milliseconds to compute PVT and protection levels, which are comparable to the average results of the cloud based solution. This result shows also that the custom embedded device performances, at computation level, can be met with a Small instance type. It's also relevant to notice that the two test environments proposed (i.e. embedded and cloud computing) can have different levels of software optimization, since the code on the embedded device is compiled to machine code, while the one in the cloud is run through a CLR interpreter. Thus the code on the embedded device can be further optimized through code rewriting and optimization, while the performances of the code in the cloud have a reduced margin of improvement in the PaaS model. These deep level optimization aspects will be the subject of future studies.

Responsiveness test: network communication

In this test, we wanted to investigate one of the possible threat to the proposed cloud based solution, which derives from moving the computation of the algorithm away from the device to the cloud platform, which introduces network communication latencies. Communication latency could in fact represent a problem for the provisioning of cloud based positioning service: if it affects response times in a measure that prevents the device from getting a position in the expected time, it could invalidate the proposed solution for most of the applications based on navigation.

In order to address this issues and test the network performances of the cloud computing platform, we have performed a set of communication tests using a 1 hour long dataset collected by our receiver. Test results show that, on over 100 requests sent from the embedded device to the cloud service and vice-versa, the measured Round Trip Time of a transaction is always under 220 ms (see Figure 11).

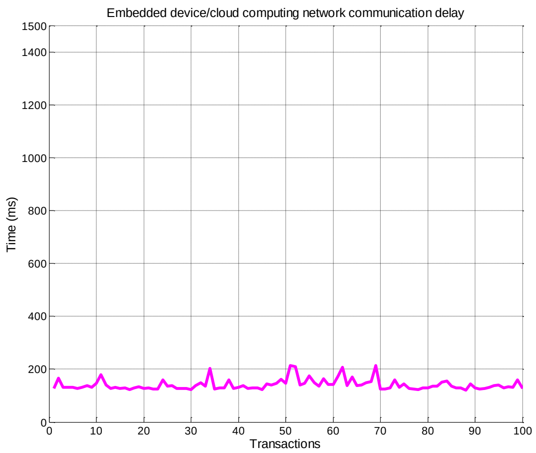


Figure 11. Device-Cloud network communication Round Trip Time (RTT)

This value, added to the algorithm execution time calculated in the previous step, indicates that our solution can still produce and deliver its expected outputs below the critical threshold of 1000 milliseconds.

Responsiveness test: access speed to cloud EDAS correction database

In this final test, we wanted to measure the time required at cloud service level to get all EDAS information, needed as input of the algorithm, from the cloud SQL database when a request from the device occurs.

In order to have a good estimation of this time, we conducted some study on the average number of EDAS RTCM messages need to have a complete set of correction to compute the augmentation: from this study emerged that, in the worst case, 140 records are needed to perform the PVT and PL algorithm.

Our tests measured the average time required to collect all the records with the reference Time-Of-Week, as sent by the device in the service request, up to the 140 previous records. As shown in Figure 12, the average time to perform this operation over 100 database transactions is around 140 ms, which is acceptable, when summed to the execution and networking time, to remain under the 1 second threshold.

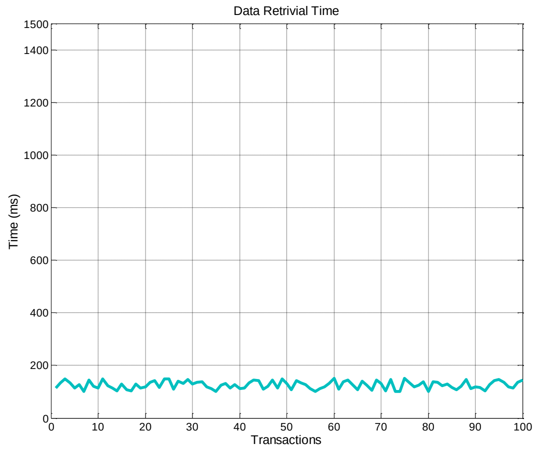


Figure 12. EDAS cloud database data retrieving delay

CONCLUSIONS

The paper presents and analyses a cloud-based architecture for centralized computation of augmentation and integrity based on EGNOS/EDAS information services. We analysed the performances of our augmentation solution and proposed a test methodology based on the definition of a set of KPIs useful to assess augmentation and integrity algorithms performances in controlled environment with post signal processing techniques. We defined a minimum set of KPIs useful to assess the quality and the responsiveness of the system comparing the output and performances of device-based and cloud-based solutions.

Tests results reveal that a cloud-based approach could represent a possible solution to provide valid augmentation services, especially in areas where poor satellites visibility inhibits the use of EGNOS corrections through SIS such as Northern European countries. Our Navigation-as-a-Service solution has emerged as a valid alternative to stand-alone augmentation positioning, insuring a good quality of positioning in term of precision and accuracy, with an high level of responsiveness (i.e. under 1 second response time). Further improvements of this study may include tests in different scenarios, both in controlled and real signal conditions. Results obtained so far demonstrate that a cloud-based solution for EGNOS/EDAS applications could be applicable in real use cases, especially for users' mass-market devices, such as smartphones and tablets. This suggests how a definition of a new set of NMEA data, which should provide more details on signal information, might be needed so that advanced positioning systems based on EGNOD/EDAS could be applied to develop new mass-market added-value Location-Based Services.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7-SPACE-2013-1) under grant agreement n° 607220.

REFERENCES

[1] GSA, "GNSS Market Report issue 3, 2013," October 2013. [Online]. Available: http://www.gsa.europa.eu/news/gsa-market-report-promises-success-gnss. [Accessed August 2014].

[2] " Data Access Service (EDAS) Service Definition Document (v2.0)," European Commission, 10/04/2013.

[3] "FLOODIS FP7 Project," [Online]. Available: http://www.floodis.eu. [Accessed August 2014].

[4] " EDAS NTRIP Service User Information Package (v2.1)," European Commission, 12/07/2013.

[5] "The Radio Technical Commission for Maritime Services (RTCM) website," [Online]. Available: http:// www.rtcm.org. [Accessed August 2014].

[6] "EDAS FTP Service User Information Package (v1.1)," European Commission, 10/04/2012.

[7] NIST, "The NIST Definition of Cloud Computing," September 2011. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf. [Accessed August 2014].

[8] "Microsoft Azure Cloud Computing Platform," Microsoft, [Online]. Available: http://www.microsoftazure.com. [Accessed August 2014].

[9] "RTKLIB: An Open Source Program Package for GNSS Positioning," [Online]. Available: http://www.rtklib.com/. [Accessed August 2014].

[10] "RTCA Minimum operational performance standards for global positioning system/wide are augmentation system airborne equipment. RTCA SC-159, DO-229D.," Washington, DC., 2010.

[11] T. Walter, "WAAS MOPS: Practical Examples," in *ION NTM* , San Diego, California, 1999.

[12] "EDAS Client Software User Manual (v03-00)," European Commission, 02/09/2013.

[13] R. T. Fielding and R. N. Taylor, "Principled Design of the Modern Web Architecture," *ACM Transactions on Internet Technology (TOIT),* vol. 2, no. 2, p. 115–150, 2002.

[14] "Visual Studio Diagnostic Tools," Microsoft, [Online]. Available: http://msdn.microsoft.com/en-us/library/dd264943.aspx. [Accessed August 2014].