

MATLAB BASED APPROACH OF DECODING EGNOS MESSAGE

Dumitru-Lucian BLĂGESCU, Ionuț-Alexandru BĂTRÎNACHE

Scientific Coordinator: PhD. Eng. Vlad Gabriel OLTEANU

University of Agronomic Sciences and Veterinary Medicine of Bucharest, Faculty of Land Reclamation and Environmental Engineering, 59 Mărăști Blvd, District 1, 011464, Bucharest, Romania

Corresponding author email: blagesculucian@gmail.com

Abstract

As the European GNSS Agency (GSA) is emphasizing its efforts in implementing EGNOS based operations in various domains, such as maritime navigation or precision farming, and as the European Commission (EC) is considering extending the EGNOS coverage to Eastern Europe, it becomes demanding to start using EGNOS at national level. A first step in such direction is decoding the transmitted messages. This paper presents a MATLAB approach of decoding different EGNOS messages types and extracting the necessary information for computing the protection levels. In order to test the decoded messages, the European Space Agency's SBAS Teacher tool will be used for comparison.

Key words: EGNOS, MATLAB,

INTRODUCTION

The European Geostationary Navigation Overlay Service (EGNOS) is a satellite based augmentation system (SBAS) developed by the European Space Agency, the European Commission and EUROCONTROL. As the current Global Navigation Satellite System (GNSS) alone cannot meet all the requirements for a reliable navigation system, it needs to be augmented by The European Geostationary Navigation Overlay Service. As currently defined, these requirements are as follows: accuracy (difference between the measured position at any given time to the actual or true position), integrity (ability of a system to provide timely warnings to users when it should not be used for navigation), continuity (ability of a system to perform its function without (unpredicted) interruptions during the intended operation), availability (ability of a system to perform its function at initiation of intended operation).

EGNOS is part of a developing multi-modal inter-regional SBAS service, which is able to support a wide spectrum of users in many different areas of applications, such as aviation, maritime, rail, road, agriculture. Similar SBAS systems have already been implemented by the US (Wide Area

Augmentation System – WAAS) and Japan (MTSAT Satellite based Augmentation System - MSAS). Development of other augmentation systems is being investigated in other regions of the world as well (e.g. GPS Aided GEO Augmented Navigation – GAGAN in India and System of Differential Correction and Monitoring – SDCM in Russia). All of these systems and their foreseen coverage is shown in Figure 1.

The EGNOS system differs from conventional GPS system in three key aspects: the provision of integrity positioning with a safety-of-life quality, a better accuracy than GPS (downs to about 1 to 2 m) and the possibility of establishing a geographical position with legal guarantees.

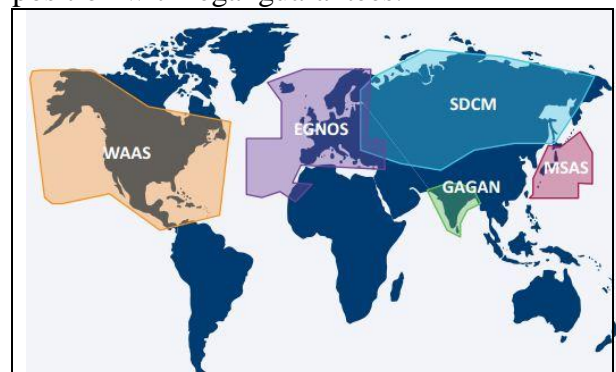


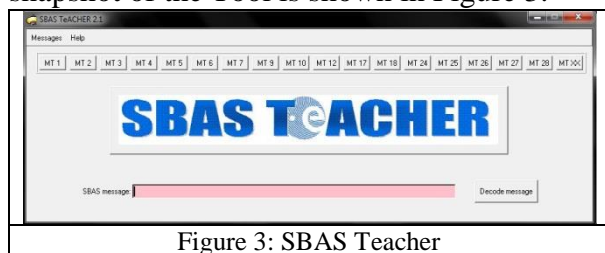
Figure 1: Worldwide SBAS systems [http://egnos-portal.gsa.europa.eu/]

In order to allow the analysis of previous EGNOS messages for different reasons, ESA has launched a service called EGNOS Message Server (EMS). Which is accessible free-of-charge, using File Transfer Protocol. EMS stores the augmentation messages broadcast by EGNOS in hourly text files, which are organized over a pre-defined hierarchy of directories (Figure 2).

Nume	Dimensiune	Data modificării
[directorul părinte]		
PRN120/		05.01.2015, 00:00:00
PRN122/		05.01.2015, 00:00:00
PRN124/		05.01.2015, 00:00:00
PRN126/		05.01.2015, 00:00:00
PRN131/		05.01.2015, 00:00:00
PRN136/		19.12.2014, 00:00:00
xferstat.dat	647 kB	01.04.2008, 00:00:00

Figure 2: EGNOS Message Server directories

Furthermore, in order to enhance SBAS understanding at user level, ESA in association with Iguassu developed SBAS Teacher, a tool which permits users to encode or decode different EGNOS message types. A snapshot of the Tool is shown in Figure 3.



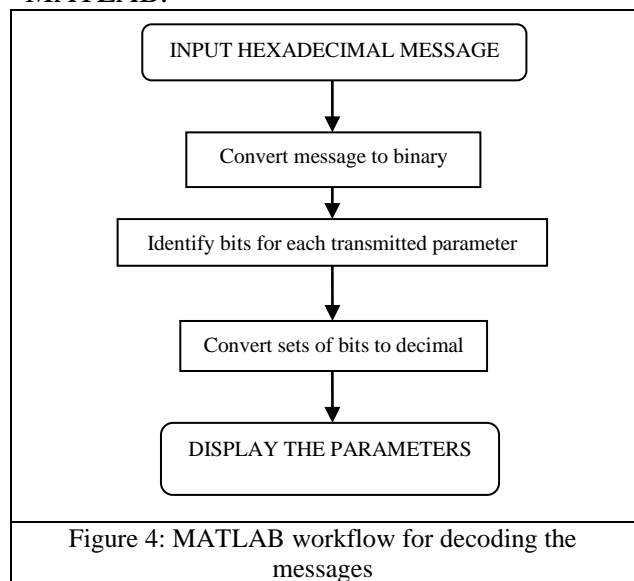
This paper describes a MATLAB approach of decoding different EGNOS messages types and extracting the necessary information as a first step in computing the protection levels and using the benefits brought by EGNOS.

MATERIALS AND METHODS

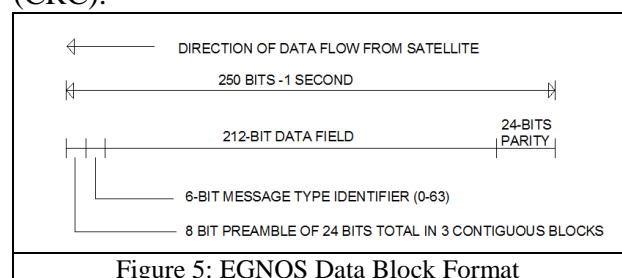
As stated before, our aim is to implement MATLAB algorithms which decode all the EGNOS message types. To accomplish this we used EGNOS Message Server (EMS) as primary data source, the European Space Agency's SBAS Teacher tool to encode different messages, and MATLAB as development platform.

MATLAB is a numerical computing environment which allows matrix manipulations, implementation of algorithms, creating of different graphical user interfaces while providing many other engineering

solutions. MATLAB users come from various backgrounds of engineering, science, and economics. And is widely used in academic and research institutions. Our goal is to develop a simple MATLAB application which decodes EGNOS messages and afterwards extract the information needed to compute the protection levels which is the main goal of EGNOS users. In this paper, attention will be focused on messages types MT6, MT9, MT25 and MT26. Figure 4 represents a diagram that contains the necessary steps to decode the messages in MATLAB.



The first step in decoding the message is to transform it from hexadecimal (as it is given in the EMS archived files) to binary, thus breaking it down into bits which can be easily handled in MATLAB. The resulting binary message is a 250 bits string (Figure 5), in which we know the first 8 bit is the preamble, the next 6 bit indicating the messages types, and the last 24 bit are assigned to the Cyclic Redundancy Check (CRC).



In this step the major difficulty consisted in identification of bit groups that needs to be decoded into the transmitted parameters as

not many information is given in the literature. The identification of these groups of bits contained in the 212 bits data field was accomplished by changing the parameters and encoding multiple messages in SBAS Teacher. Afterwards the messages were decoded in MATLAB to observe in the binary string which group of bits is altered. Through several attempts, and based on available information we accomplished to decode the MT's mentioned above.

The decoding was performed by repeated attempts because there is no free of charge documentation available on the decoding algorithm. The only resource that we relied on was a presentation about EGNOS done by Research group of Astronomy and Geomatics (gAGE/UPC) from Polytechnic University of Catalonia)

MT6 (Figure 6) contains integrity information for 51 satellites, which is the maximum number of satellites that can be present in the PRN mask. This message also includes IssueOf Data Fast Corrections (IODFj) (j=2...5) to relate the User Differential Range Estimate Indicator (UDREI) to the fast corrections included in messages of type 2 to 5 or 24.

MT6 can be used in two different ways: on one hand, it allows the fast corrections to be updated infrequently while on the other hand, it may be also used in case of satellite alert conditions. It has to be remarked that this message does not include an Issue Of Data PRN Mask (IODP) and hence the link to the PRN mask is not provided within the message. The UDRE indicators included in MT6 do apply to the satellites defined in the last received PRN mask.

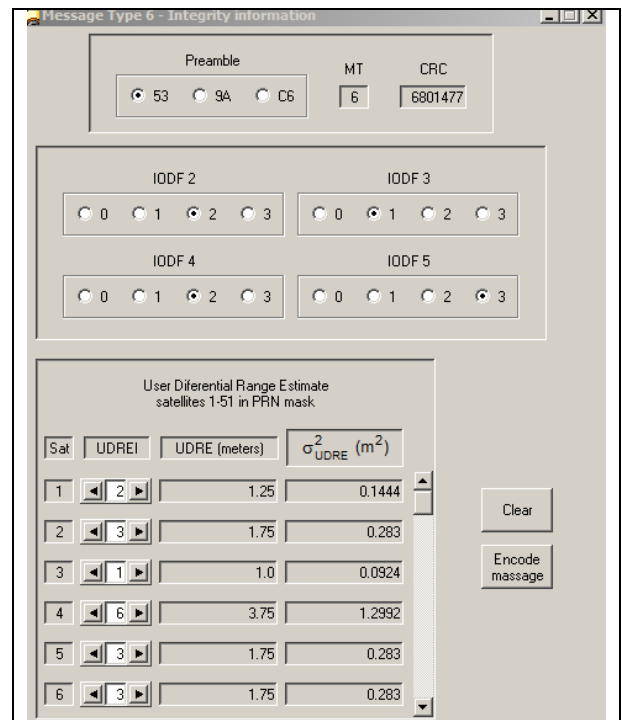


Figure 6: Message type 6-Integrity information

Message type 9 contains the information about the GEO navigation (Figure 7). As GEO satellites do not belong to any global or regional satellite positioning system, ephemeris for those satellites are not available to the users. Therefore, it is the SBAS that is in charge of providing the user with the GEO ephemeris (“The EGNOS SBAS Message Format Explained”, Daniel Porras Sánchez & César PisoneroBerges, GMV S.A).

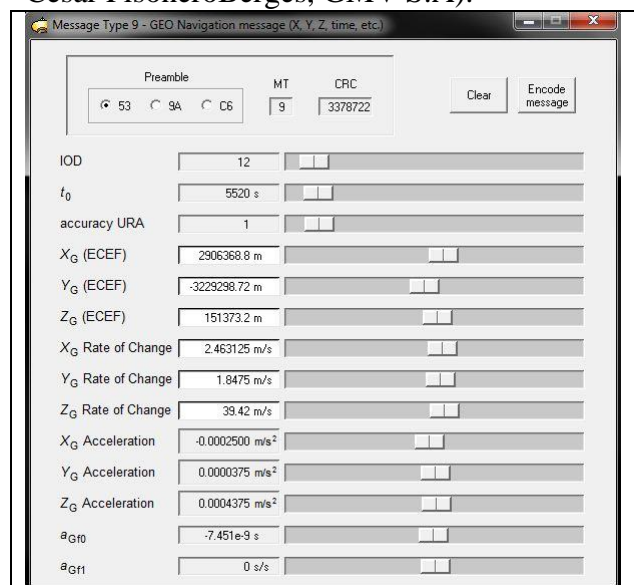


Figure 7: SBAS TeACHER Message Type 9 - GEO Navigation parameters

Message 25 provides corrections for satellite ephemeris and clock errors (Figure 8).The

data block is divided into two parts. The first bit of each part is the velocity code, which if it's set to 1 indicates that in that part of the message clock drift and velocity corrections are included, otherwise only the clock offset and position corrections are provided, but for two satellites instead of 1. Therefore the message can consist of corrections for 1, 2, 3 or 4 satellites, depending on the values of the velocity bits and how many satellites are being corrected ("The EGNOS SBAS Message Format Explained", Daniel Porrás Sánchez & César PisoneroBerges, GMV S.A).

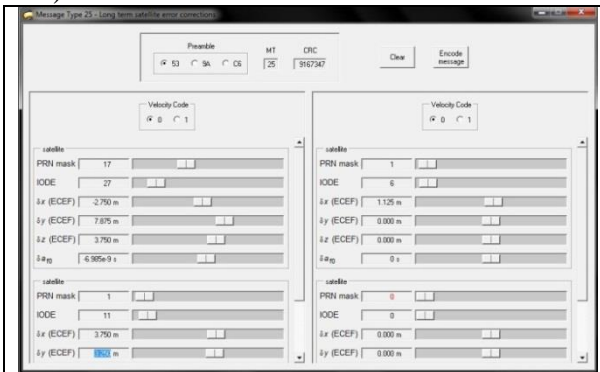


Figure 8: SBAS TeACHER Message Type 25 – Long term satellite error corrections

Message type 26 provides the vertical ionospheric delays and their accuracies for the IGP's identified by the band number and the IGP number (Daniel Porrás & Cesar Pisonero, 2006).

RESULTS AND DISCUSSIONS

For decoding the MT6 we encoded with SBAS Teacher a simple message multiple times, changing each time the same parameter until we could identify the bits which define that specific parameter. In this example we will choose to find the IODF 2 parameter and try to isolate the group of bits that define it.

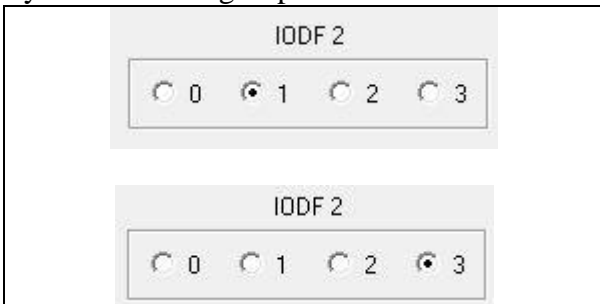


Figure 9: IODF 2 for MT6

In Figure 9 MT6 was encoded once with the IODF 2 value set to "1" and once with the IODF 2 value set to "3". After the hexadecimal string generated by SBAS Teacher was decoded using our MATLAB application we searched in the binary strings for "01" (the binary equivalent of decimal "1") and "11" (the binary equivalent of decimal "3") (Figure 10)

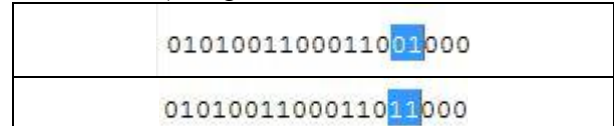


Figure 10: IODF 2 bits identified in the binary string

The same algorithm was used to decode the rest of the message, as well as for the other message types.

For the MT9 the message was converted into binary in MATLAB and obtained a 250 bits string, a user friendly form to work with. A small part of it may be seen in Figure 11. In Figure 12 one can observe a side by side comparison between the parameters decoded with the MATLAB program and the same parameters encoded with SBAS Teacher.

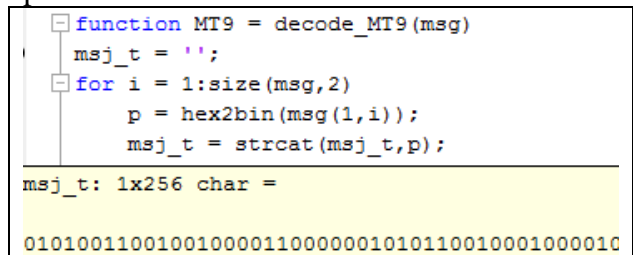


Figure 11: Message Type 9 decoded in a sequence of ones and zeros

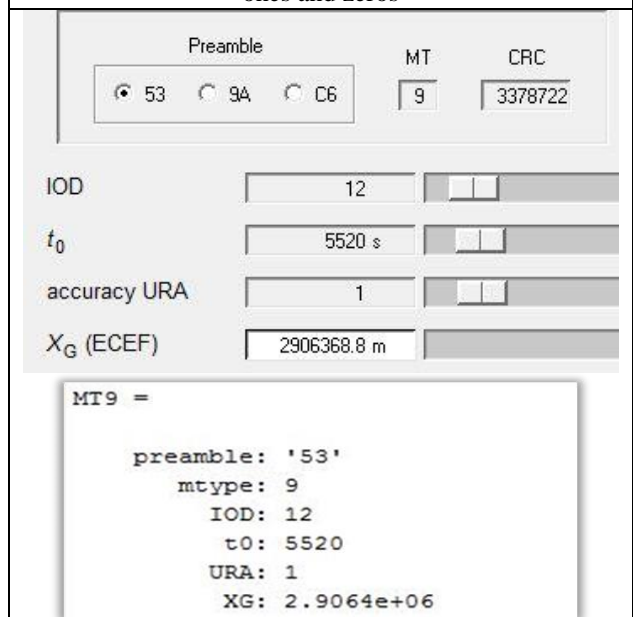


Figure 12: Comparison between MATLAB and SBAS

Figure 13 and Figure 14 present the results of the EGNOS messages decoding application for MT25. Comparing these results with the parameters from the Figure 7 it can be seen that the numbers are similar, so the decoding algorithm is working properly.

```

MT25 =

  preamble: '53'
  mtype: 25
  velocity: 0
  prn1: 17
  iode1: 27
  delta: [1x1 struct]
  prn2: 1
  iode2: 11
  IODP: [1x1 struct]
  velocity1: 0
  prn3: 1
  iode3: 6
  prn4: 0
  iode4: 0
  Parity: '1000101111110000111110011'

```

Figure 13: The results of the MATLAB decoding program for message 25

```

delta.x1=-2.75
delta.y1=7.875
delta.z1=3.75
delta.af01=-6.98492e-08

```

Figure 14: The results of the MATLAB decoding program for message 25

To test the reliability of the decoding program, a series of different messages were generated using SBAS Teacher and decoded using the following MATLAB decoding program.

```

function MT26 = decode_MT26(msg)
msj_t = '';
for i = 1:size(msg,2)
    p = hex2bin(msg(1,i));
    msj_t = strcat(msj_t,p);
end
%Preambul
MT26.preamble =
strcat(num2str(bin2dec(msj_t(1:4))),n
um2str(bin2dec(msj_t(5:8))));
%Tip mesaj
MT26.mtype = bin2dec(msj_t(9:14));
% Band Number
MT26.band = bin2dec(msj_t(15:18));
% Block ID
MT26.block = bin2dec(msj_t(19:22));
% IGP Vestical Delay Estimate
MT26.deley=zeros(15,1);
for i=1:15
    MT26.deley(i,1) =

```

```

bin2dec(msj_t(23+13*(i-1):23+13*(i-
1)+8))*0.125;
end
% Grid Ionopheric Vertical Error
Indicator (GIVEI)
MT26.GIVEI= zeros(15,1);
MT26.GIVE= zeros(15,1);
MT26.sigmaGIVE = zeros(15,1);
for i = 1:15
    MT26.GIVEI(i,1) =
bin2dec(msj_t(32+13*(i-1):32+13*(i-
1)+3));
    [MT26.GIVE(i,1),
MT26.sigmaGIVE(i,1)] =
chooseGIVEI(MT26.GIVEI(i,1));
end
% Issue of Data
MT26.issue = bin2dec(msj_t(218:219));
end

```

MT26 =

```

  preamble: '53'
  mtype: 26
  band: 3
  block: 0
  deley: [15x1 double]
  GIVEI: [15x1 double]
  GIVE: [15x1 double]
  sigmaGIVE: [15x1 double]

```

IGP	GIVEI	GIVE (meters)	σ_{GIVE}^2 (m ²)	
1	1	0.6	0.0333	0.250 m
2	2	0.9	0.0749	1.250 m
3	3	1.2	0.1331	5.000 m
4	4	1.5	0.2079	8.875 m
5	5	1.8	0.2994	9.875 m

Preamble: 53 9A C6 MT: 26 CRC: 5678742

Band number: 0 1 2 3 4 5

Block ID: 0 1 2 3 4 5

GIVEI=1	GIVE=0.6	sigmaGIVE=0.0333
GIVEI=2	GIVE=0.9	sigmaGIVE=0.0749
GIVEI=3	GIVE=1.2	sigmaGIVE=0.1331
GIVEI=4	GIVE=1.5	sigmaGIVE=0.2079
GIVEI=5	GIVE=1.8	sigmaGIVE=0.2994

Figure 15: Comparison between SBAS and MATLAB decoding

