Simulation Programs for the Evaluation of the LDPC-Coded Multicarrier Transmissions

Dr. Vasile Bota, Mihaly Varga Ph.D. student, Dr. Zsolt Polgar
Data Transmissions Laboratory, Communications Department, Technical University of Cluj-Napoca
Email: Vasile.Bota@com.utcluj.ro

The set of programs that simulate the LDPC-coded multicarrier transmissions (MC) is composed of two programs:

- MP_SimSU employed for the performance evaluation of LDPC-MC for a point-to-point or a single user connection;
- MP_siteSim intended for the performance evaluation of a multi-user downlink connection in a mobile environment.

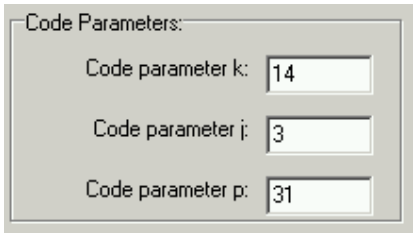
MP_SimSU

The MP_SimSU program simulates the LDPC-coded or non-coded MC transmissions. This program allows the performance evaluation of the MC-type systems OFDM, for the radio channel, and xDSL, for the „copper" channel.
For a high bit-rate transmission, the MC systems divide the available frequency band into more channels with a narrower bandwidth, while the input data flow is divided in the same number of data flows lower bit rates. These flows are transmitted independently using QAM modulations in the channels resulted by the division of the available frequency band. This approach ensures a high bit rate at a relatively low bit-error rate. Besides, it has other advantages, compared to the single carrier approach, advantages that are outlined in the pertinent literature.
The block diagram of the processing performed by the MP_SimSU is depicted in figure 1.

Brief Description of the Component Blocks

The ***Data Generator*** block generates pseudorandom test data and divides it into a data flows that is to be LDPC-coded and a data flow that is be transmitted non-coded, according to the pattern inserted in a edit-box which will be described below. The bits that are to be coded are delivered to the encoding block, and the bits are to be sent non-coded are delivered directly to the bit-loading block. The program user should set the length of the test sequence.

The ***LDPC-Encoder*** block encodes the input data flow with a systematic block LDPC code. The simulation program allows the employment of any type of array-based regular LDPC code. The control and the encoding matrices are built off-line by the encoding block, according to the three parameters k, j, p that define this type of LDPC codes. The encoding block adapts, by shortening, the encoding matrix (and the codeword length) according to the number of coded-bits that could be mapped in a symbol period or into a user bin; this number depends on the number of tones and symbol periods (time-frequency spreading) and on the mapping configuration, both set by the user. At this stage the user has to define the LDPC code by setting the three parameters, see the edit- box below.



Edit box for setting the LDPC code

The ***Simulation Parameters*** block takes the following parametrs that are set by the user:

- the test length;
- the maximum numberof iterations that would be performed by the Message-Passing decoding algorithm of the LDPC codes;
- the slope of the attenuation vs. frequency characteristic of a cable (only for the xDSL simulations) in dB/tone;
- the difference between the noise variance value stored in the soft-demapper and the actual value of the noise variance (SNR dif. in dem. [db]);
- the minumum and maximum values of the SNR between which the simulation is performed, and the step of variation of the SNR. The corresponding edit box is presented under figure 1

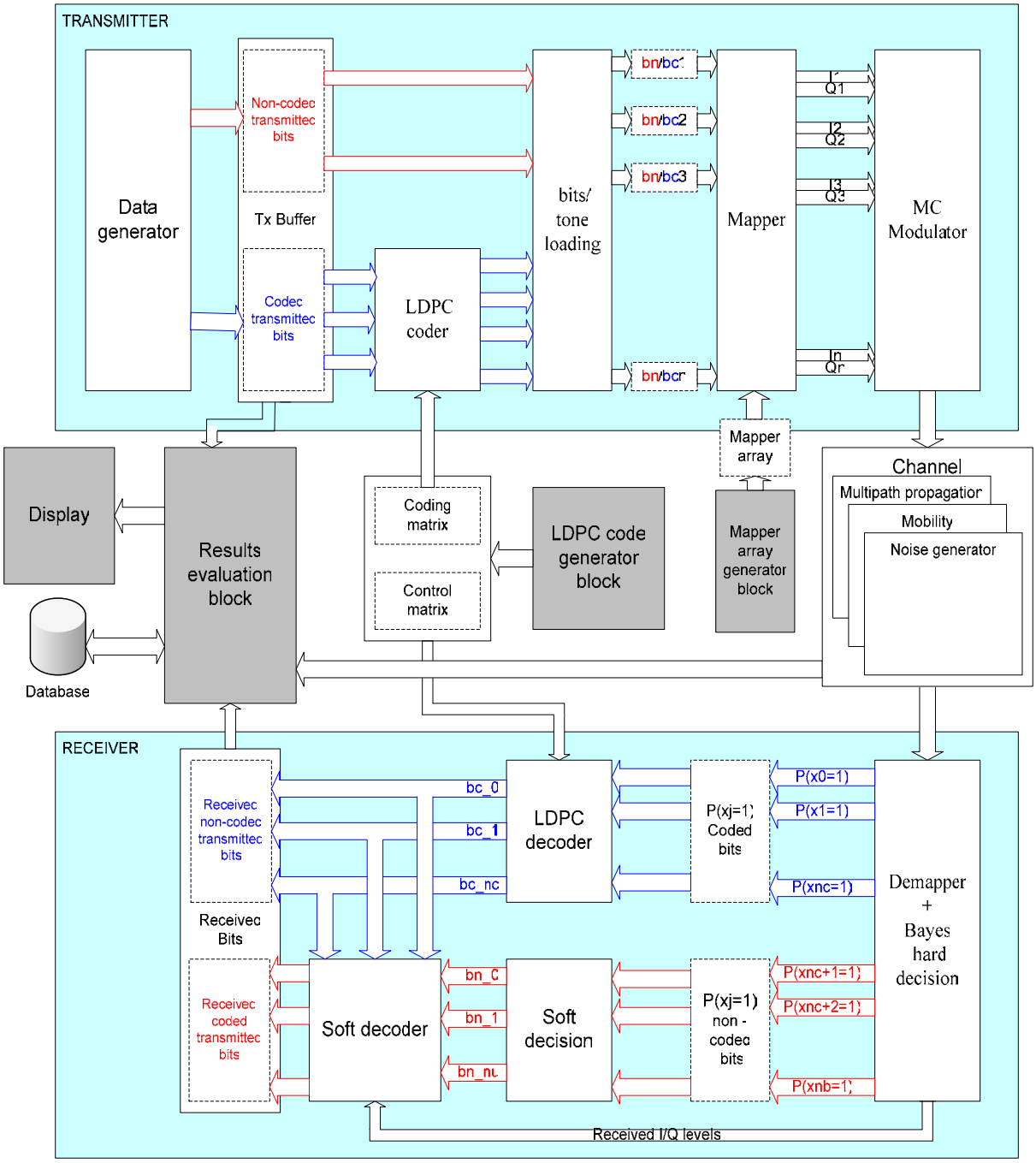
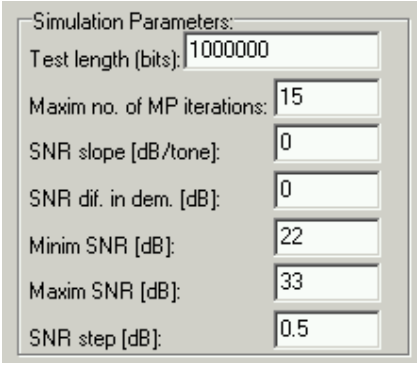
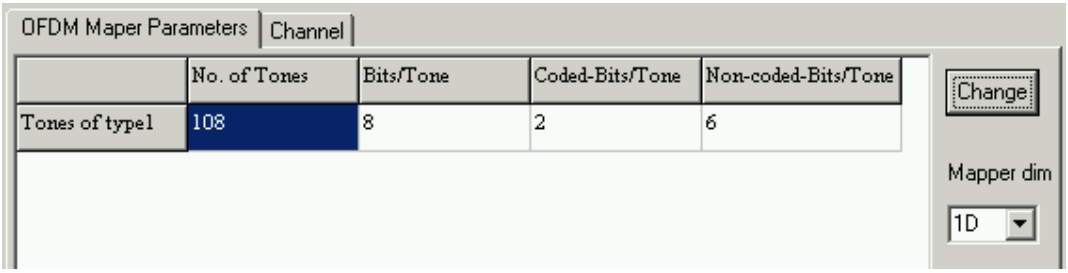


Figure 1. Block diagram of the MP_SimSU simulation program



Simulation Parameters edit box

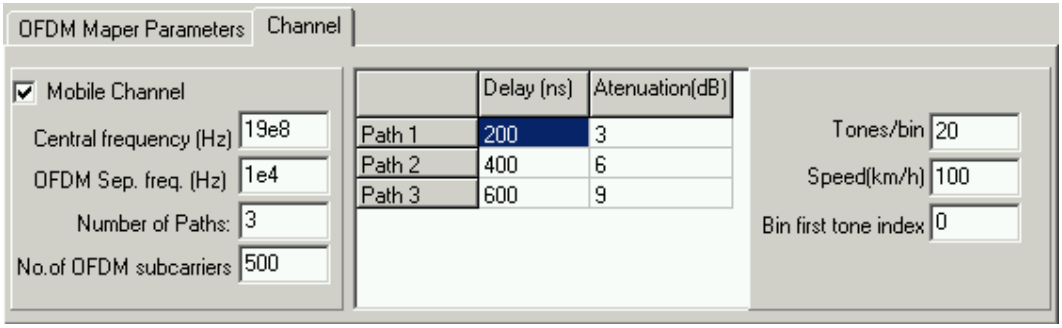
The ***Bit-loading*** block splits the input data stream into more smaller bit rates data flows that are to be loaded on each tone. This block assembles the bit sets that are to be sent in the next symbol period (or on the next user bin) on each tone. The bit-loading may be defined for each tone, for groups (types) of tones or for all tones. With the xDSL systems the magnitude of these sets may vary for different groups of tones according to the cable's attenuation characteristic.
This block may combine coded and non-coded bits according to the loading pattern selected by the user for each group (type) of tones in the edit box below.



Bit-loading edit box

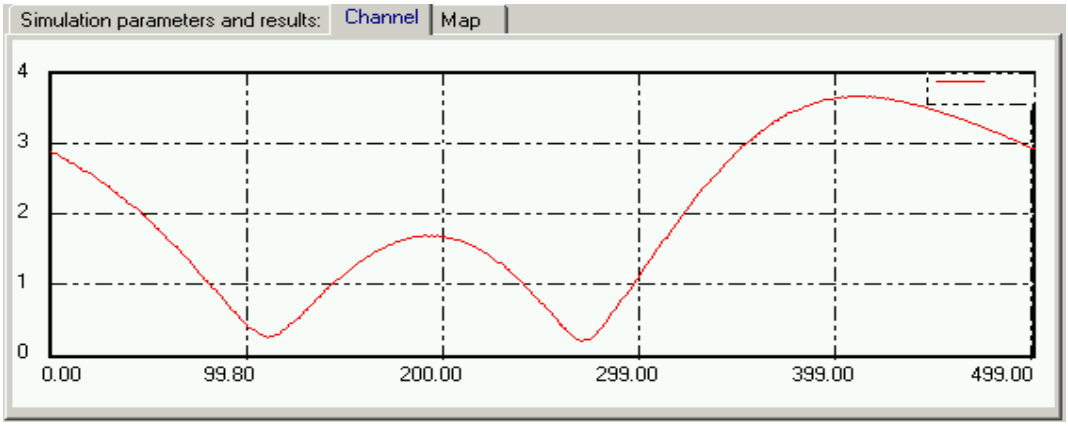
The ***Mapping*** block computes the modulating levels of each coordinate of the QAM points, according to the multibits delivered by the bit-loading block. It employs a 2-level Gray mapping for the mapping of the coded and non-coded bits of the multibit. See the edit box above.

The ***Modulator*** block performs the actual modulation of the subcarriers. The user should specify the total number of subcarriers (the values inserted in the previous ***Bit-loading*** edit box should match the value inserted here!) the frequency separation between subcarriers and the central frequency of the available bandwidth. The frequency spreading of a user-bin and the index of the first tone of the user-bin should also be defined here.

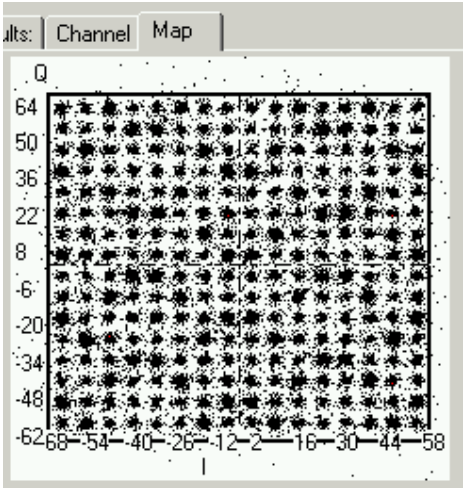


Channel parameters and user-bin setting edit box

The ***Channel Simulator*** block performs the simulation of the radio fixed or mobile channel, for OFDM-MC transmissions, and the simulation of the cable channel for the DMT-MC transmissions.
The cable channel is simulated by the slope of the attenuation characteristic (see Simulation Parameters Edit box). The impulse noise generator is in preparation and would be added to a later version of the program.
The fixed or mobile radio channel is selected by setting the speed. The Rayleigh faded multipath propagation is employed for the simulation of the radio channel. The user should set the number of the propagation paths and their attenuation and delay referenced to the first arrived path.
A Gaussian noise is superimposed upon the received signal, both for radio channel and cable channel; the range of the SNR and the step of variation of the SNR within this range are set in the Simulation Parameters edit box.
The program displays dynamically the attenuation characteristic of the channel and the position of the currently received vectors in the I-Q rectangular axes, see the figures below.



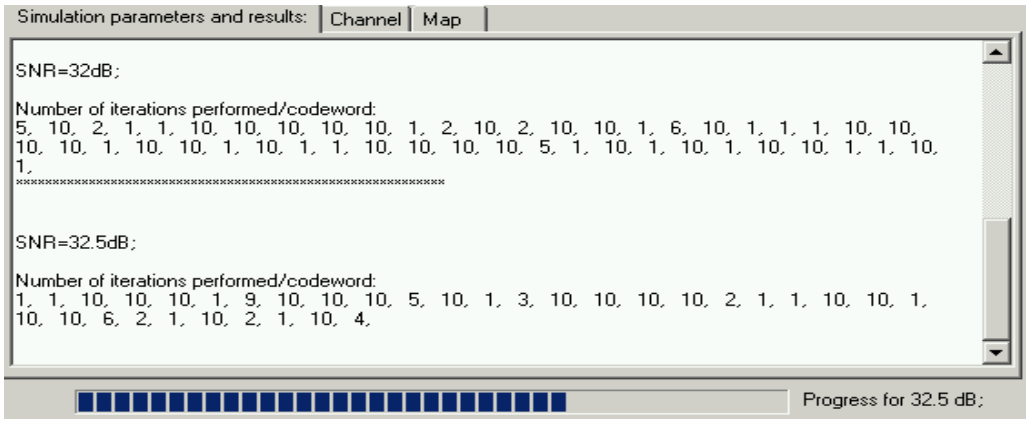
Channel window -Channel attenuation characteristic in terms of the subcarrier index.



Map window - Positions of received vectors for a 256-QAM transmission

The ***Demapper*** performs the soft-demapping of the received vectors separately on each coordinate. It computes the a posteriori probabilities of each bit in terms of the vectors received during an OFDM symbol period. This block also separates the coded bits and non-coded bits that were loaded on each symbol (bit-unloading).

The ***LDPC-decoding*** block decodes the coded bits using the Message-Passing decoding algorithm and extracts the information bits that were LDPC-coded. The user has to set the maximum number of iterations that would be performed the MP decoding algorithm, see the Simulation Parameters edit box. If the codeword is not corrected in the allowed number of iterations the program considers the codeword in error and delivers it as obtained after the last iteration. The number of iterations performed on each codeword is displayed in **Simulation parameters and results** window, indicating the SNR value as well.



Simulation parameters and results window

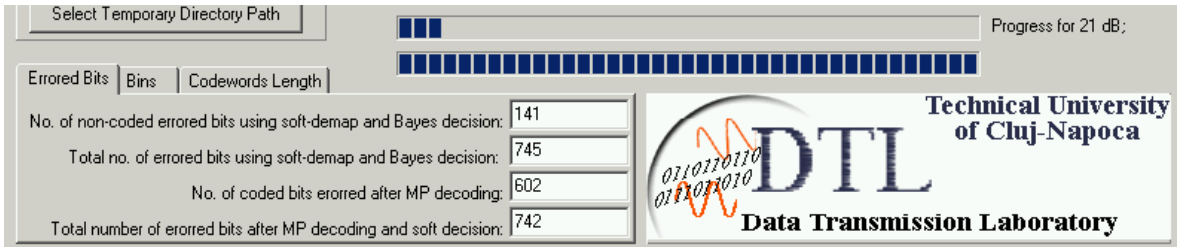
The ***Hard (Bayes) Decision*** block performs a hard decision of the coded and non-coded bits using the Bayes criterion. The hard-decided bits are employed only as a reference to evaluate the performances of the MP decoder (for the coded bits) and of the soft decision block, for the non-coded bits.

The ***Soft Decision*** block performs a soft decision of the non-coded bits employing the received coordinates and the decoded bits provided by the MP decoder. It also employs the 2-level gray mapping performed in transmitter.

The ***Results Evaluation*** block compares the received (decoded and decided) bits to the transmitted bits and stores in the data base the following information that would be used for the analysis and evaluation of of the performaces of the simulated transmission scheme:

- Informations concerning the simulation scenario (the LDPC-code the mapping table, the channel parameter, the simulation length in information bits, etc.);
- The total number and position of the error bits after the hard (Bayes) decision;
- The number and positions of the error non-coded bits after the Bayes decision;
- The number and positions of the error coded bits after the MP decoding;
- The number and positions of the error non-coded bits after the soft decision (these last two values are useful to establish the type and aparameters of an outer code in a concatenated- codes error-correcting scheme);
- The total number of the error bits, both coded and non-coded, after the MP decoding and soft-decision
- The number of iterations performed by th MP decoder for each codeword;
- The number of error bins or OFDM (DMT) symbols.

Some of these values are dispalyed during the simulation in the corresponding display boxes, see figure below.

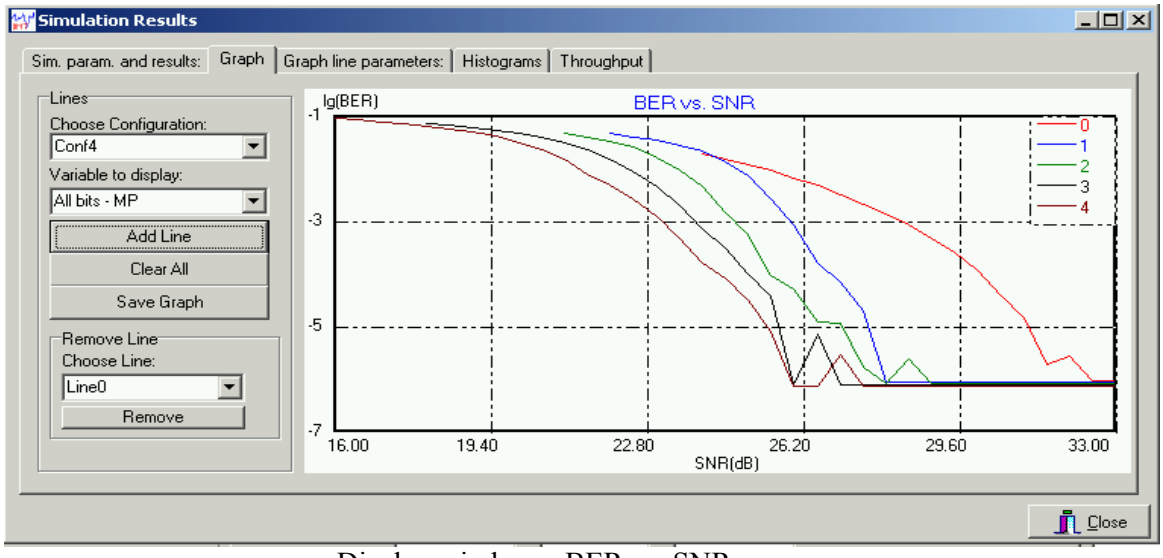


Number of error bits and bins displayed during the simulation

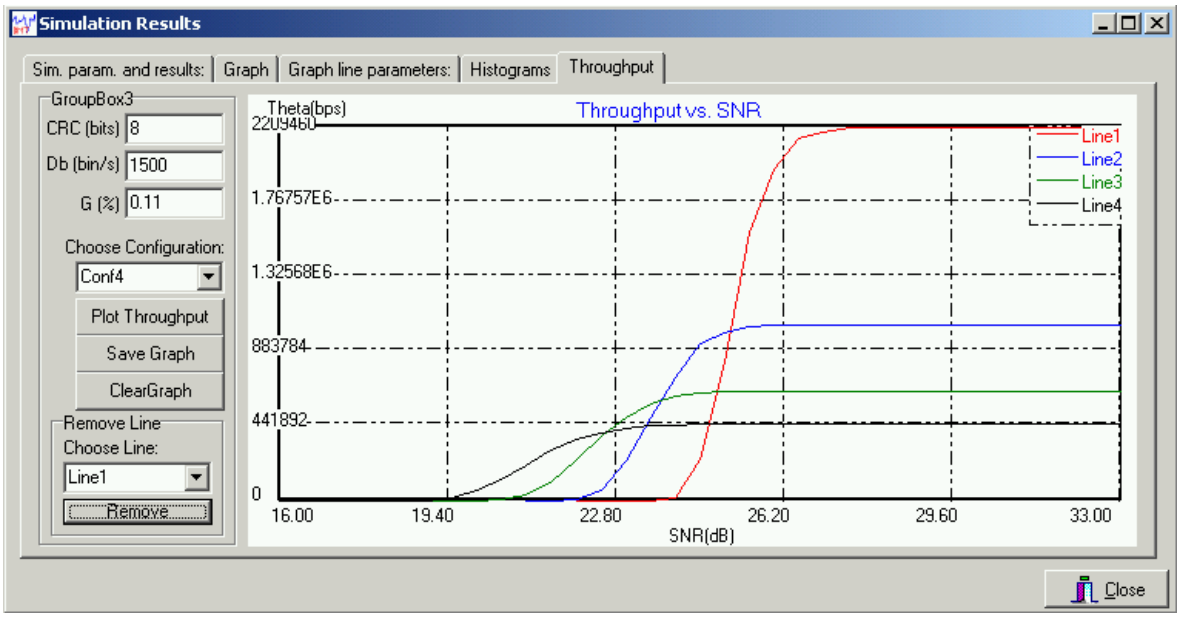
The ***Results Display*** block allows the display of the results as graphs or histograms, represented for the selected SNR range and step. The following results may be shown in the dispaly window for the non-coded or coded scheme defined:

- The BER vs. SNR curves of coded, non-coded bits or of all bits before the MP decoding and soft decision (Bayes decision) or after them.
- The bin or OFDM (DMT) symbol error vs. SNR before the MP dcoding and soft decisions or after them.
- The throughput vs. SNR.

The BER vs. SNR and throughput vs. SNR curves of five tranmission configurations as displayed in the Display Window are shown in the two figures below.



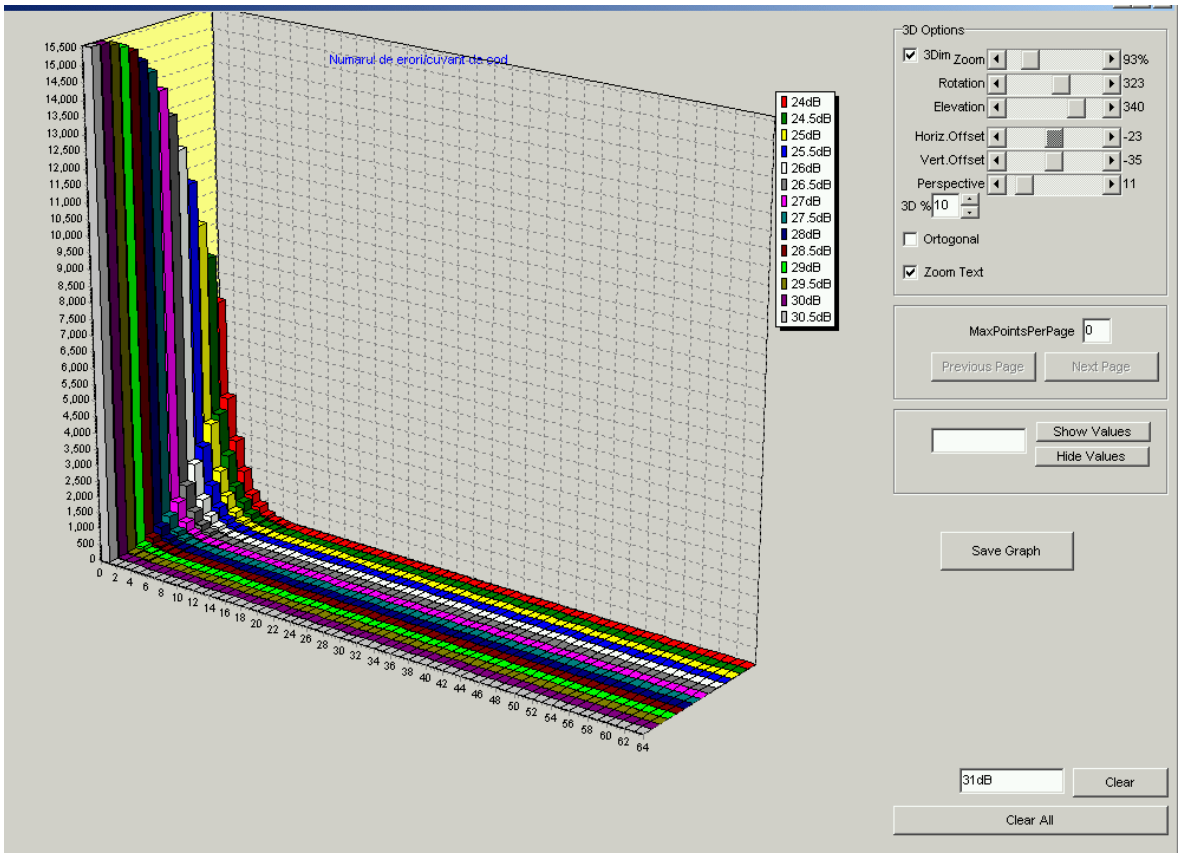
Display window – BER vs. SNR curves



Display window – throughput vs. SNR curves

Two performances of the simulated scheme may be visualized as tridimensional histograms, namely:

- the occurence frequency of bit-errors within a settable number of bits vs. SNR, before or after the MP decoding and soft decision, see the figure below; this might be useful for the choice and dimensioning the outer code in concatenated-code error-correction scheme;
- the distribution of the number of iterations performed by the MP decoder vs. SNR.



Dispaly window - the occurence frequency of bit-errors within blocks of 64 bits vs. SNR

For a faster running of the simulations the MP_SimSU application may be distributed in a TCP/IP network. The block diagram of the distributed application is shown in figure 2.

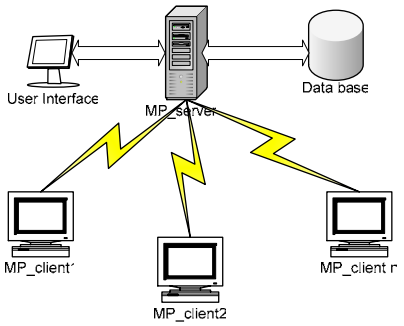
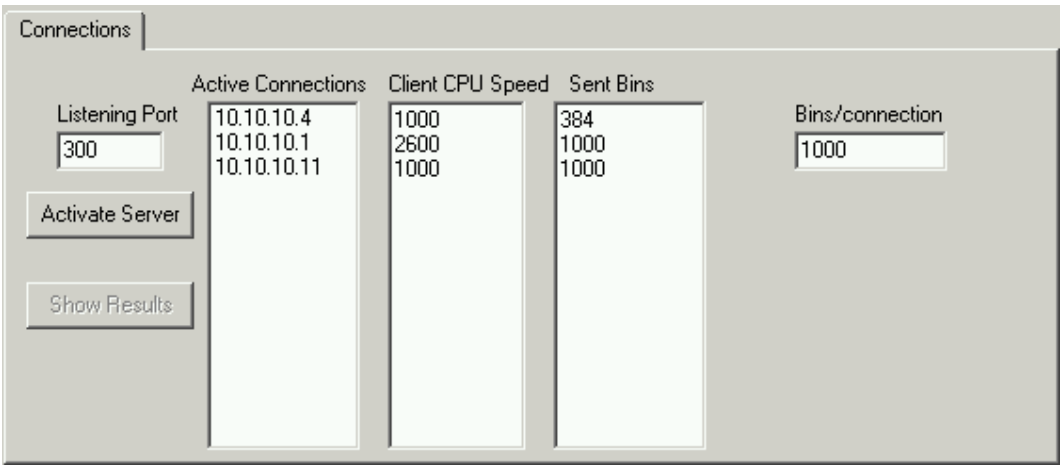


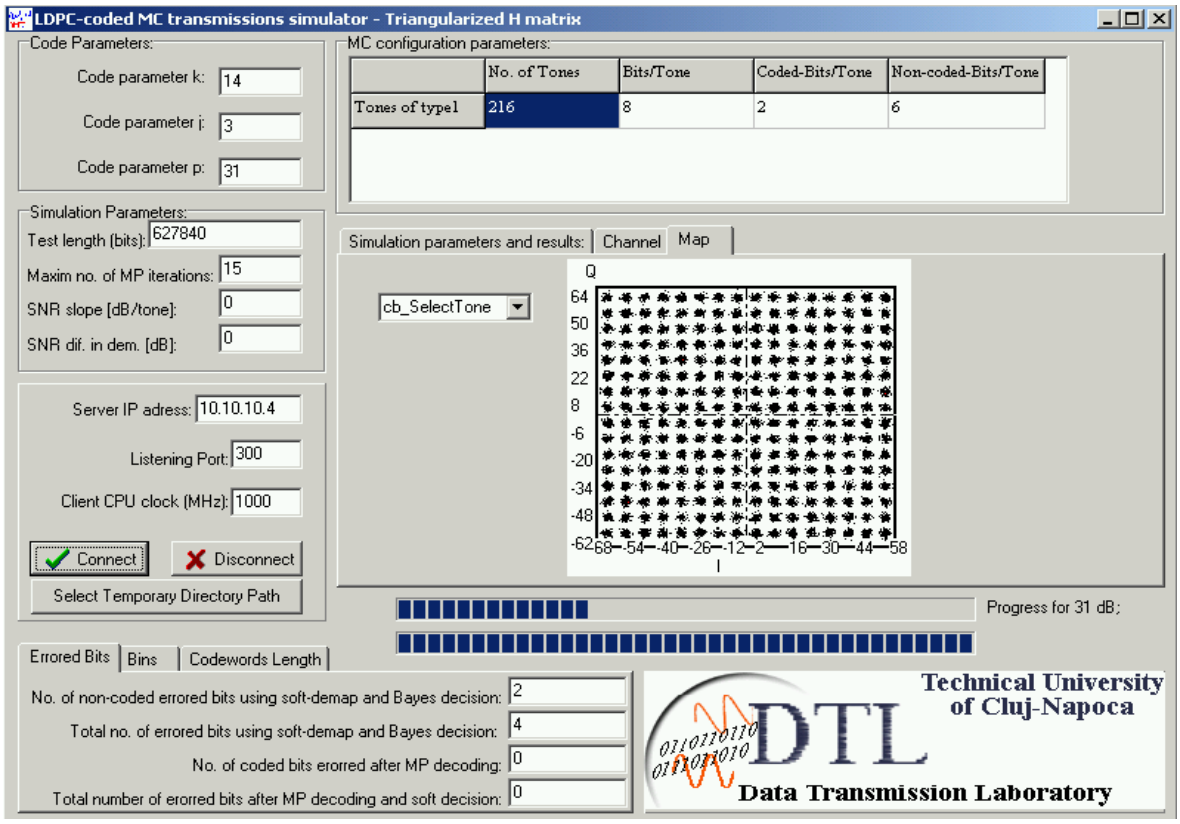
Figure 2. Block diagram of the distributed MP_SimSU application

The MP-server application comprises the user-interface, the database interface and the blocks that perform the off-line processing, marked with grey in figure 1. Additionally, it contains process-control block and an interface for communication with the client applications.
When starting the server application, the user configures the desired simulation and indicates the port to be used by the system for communication, so that the client applications could be connected to the server application by a TCP/IP network.
During the connection to the server application, the client application sends to the server the parameters of the computer it runs (IP address, processor frequency, etc). Consequently, the server application opens a new thread for each client, which ensures the communication with that client and records the tasks of the respective client. The tasks are shared between the connected clients according to their processing capability, so that the time required by each client to fulfill its tasks would be about the same. After completing the received tasks, each client delivers the results to the server, which performs the required processing and stores them in the database. The server window that contains the client's connection parameters is shown below.



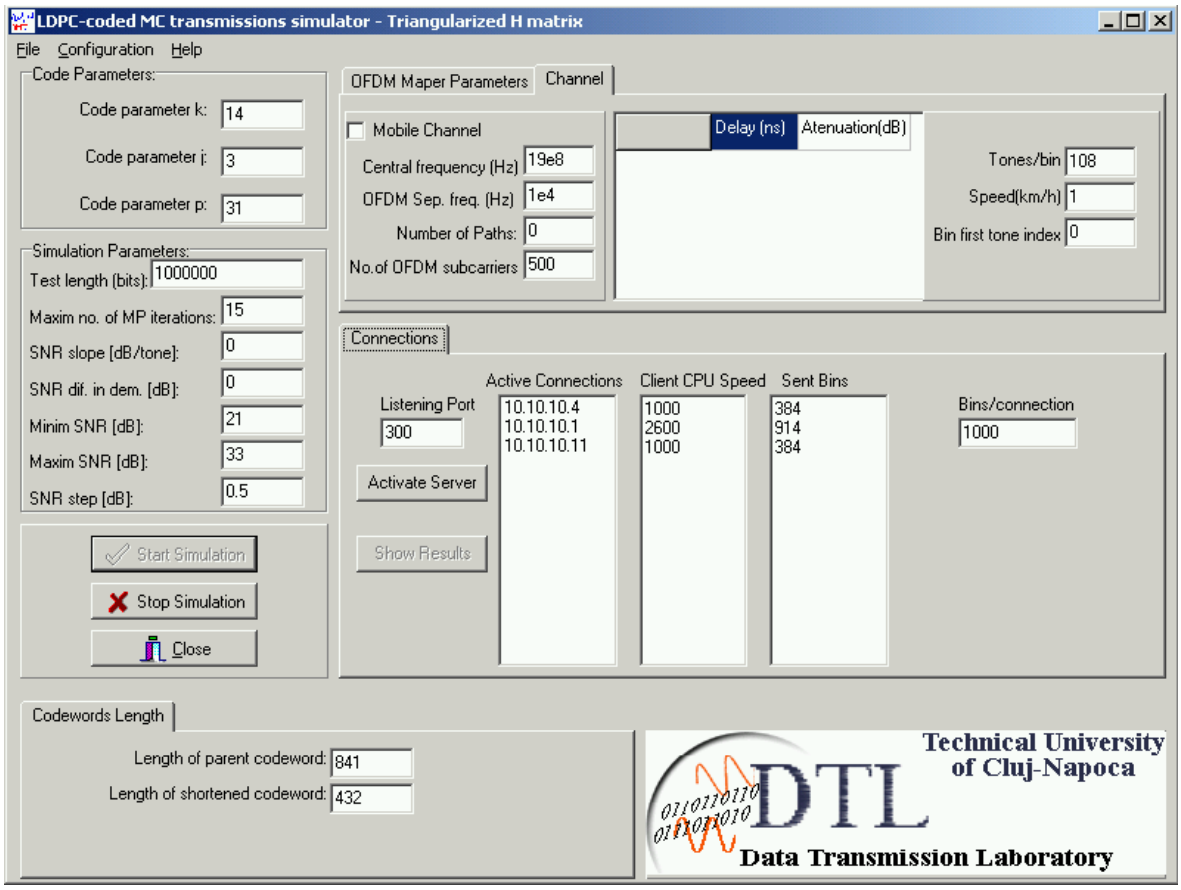
Clients connected to the MP_server and their parameters

The user-interface of the MP_client application is shown in the figure below



MP_client – User-interface

The user interface of the MP_server application, during the simulation, is shown the figure below.



MP_server – User-interface

The ***Configuration*** pull-down menu allows the selection of the coded or non-coded MC scheme and the selection, for coded configurations, of two types of array-based LDPC codes. It also allows the building and employment of preset configurations. The preset configurations contain the parameters of more configurations (LDPC code, QAM constellation, etc.) that are stored in a .txt file. The programs runs the simulations for all the configurations included in that file and may display the resulting curves on the same graph.
The ***File*** pull-down menu contains the file-type facilities: saving graphs, saving files with results and setting values or loading preset configurations.
The MP_SimSU application simulates the MC transmission, LDPC-coded or non-coded, for one station (user) connected to the base station. It may as well be employed for simulating the point-to-point OFDM transmissions on fixed radio channels or for DMT transmissions for the subscriber connection in A(V)DSL systems. The results obtained allow for the selection of coded modulation schemes (configurations) for a given channel scenario and the establishing of the thresholds between the SNR ranges where those schemes are optimum.

MP_siteSim

The simulation MP_sim set of programs contains a second program which simulates the downlink connection of a multi-user site in a mobile environment. The uplink connection is assumed to be ideal. In this case the channel model may be set differently for every user (speed, attenuation). So, every mobile station within the site has a rather different channel model.
Every MS sends (on the uplink, which is assumed to be ideal) the average attenuation characteristic of its channel (averaged on every possible bin position). Based on this data, the scheduler in the base station allocates its user the best available bin position and selects the configuration code-modulation that might ensure the highest throughput.
This updating, done for every bin, selects the optimum (coded) configuration from a stored set of configurations. The set of (coded) configurations and the thresholds were selected based on the simulations performed with the single-user MP_Sim SU program and are inserted by the program user before the simulation.
Figure 3 shows the tasks of each of the two simulation programs.

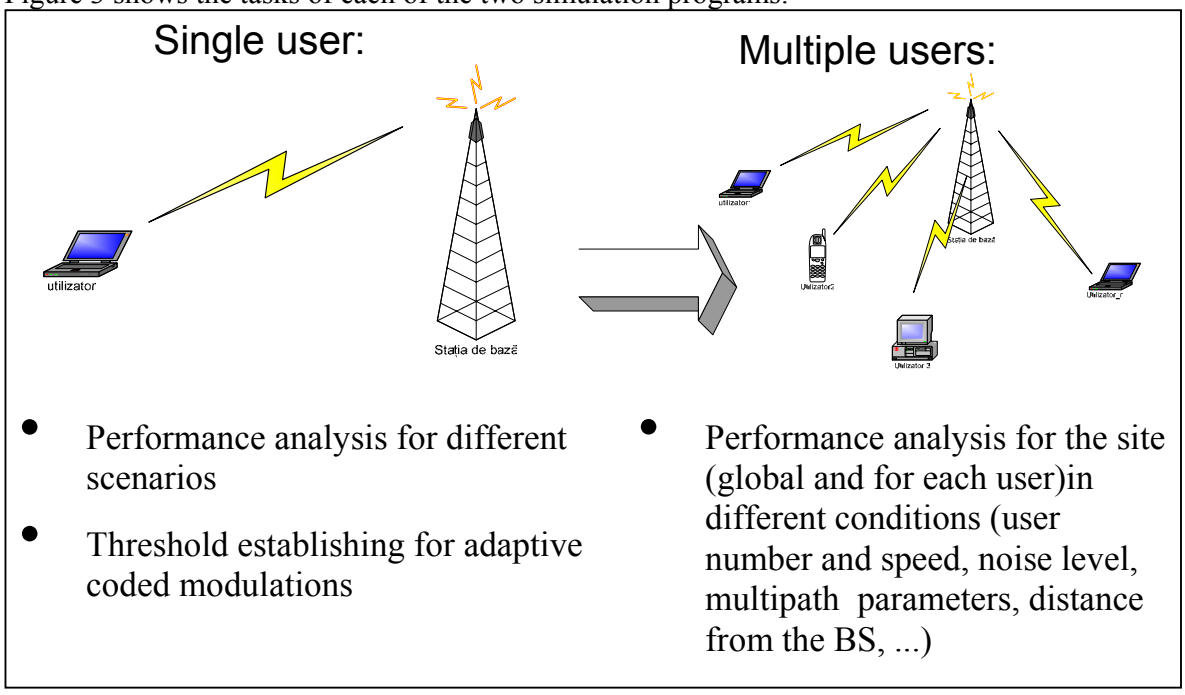
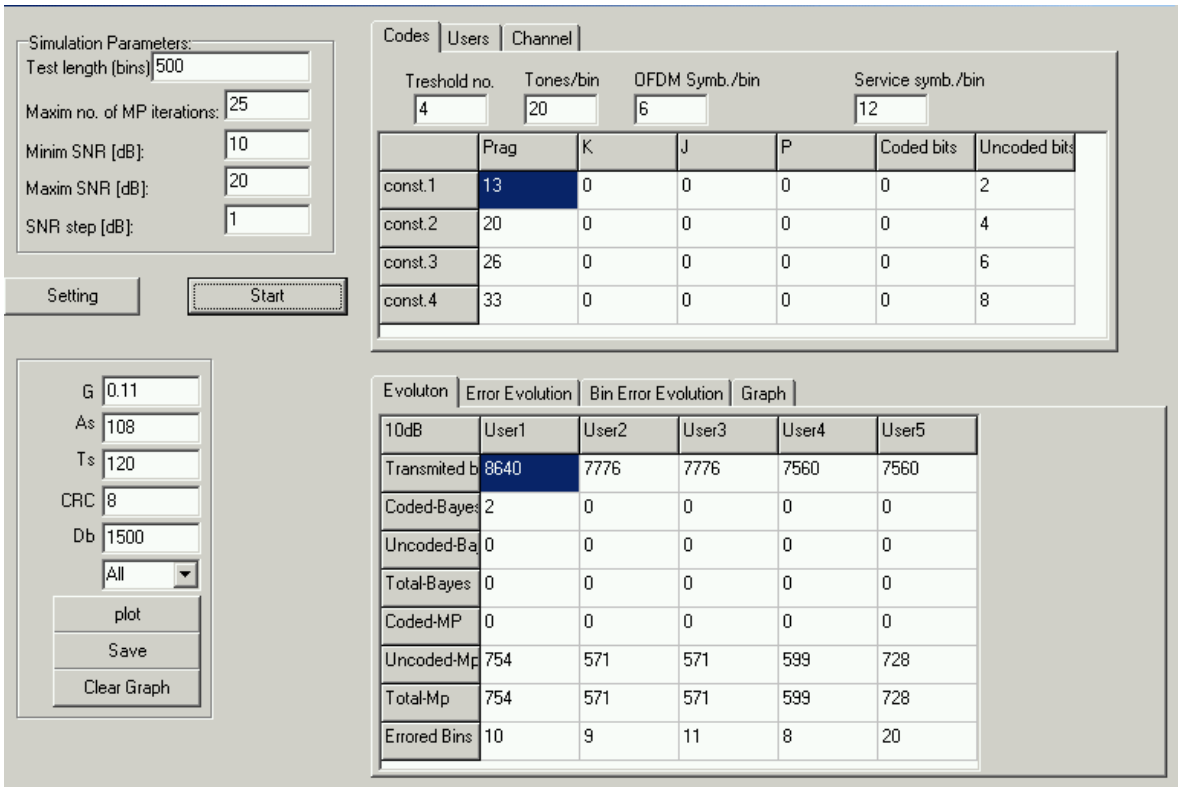


Figure 3.

In this application the user has to set:
- The configurations code-modulation that form the stored family;
- The SNR threshold between which each configuration should be used;

- The number of multiple paths and their relative delays and attenuations; these are common for the whole site;
- The number of active mobile stations;
- The speed and attenuation of each station.

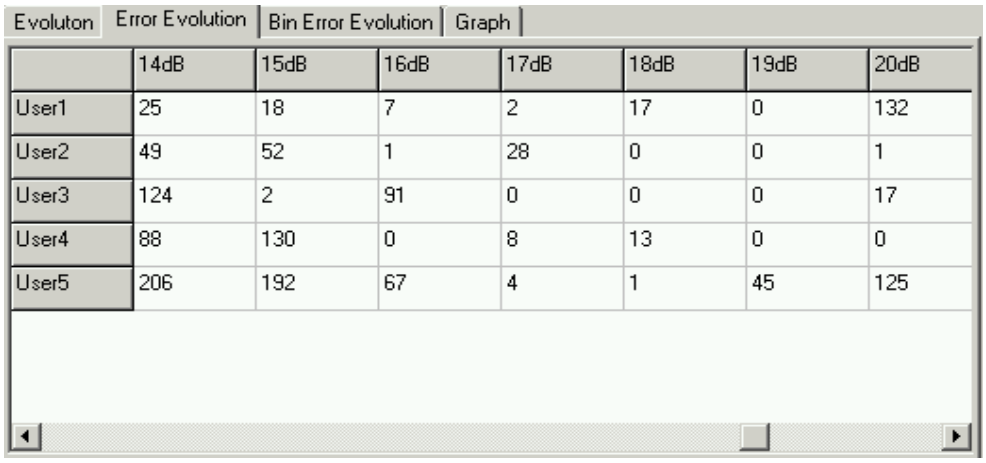
The user-interface of this application is shown in the figure below.



User-Interface of the MP_siteSim application

The block diagram of the MP_siteSim is presented in figure 4. The program performs about the same processing as the previous program, see figure 1, but the processing is performed in parallel for each user. There should be noted that the MP_siteSim does not employ a single coded configuration, but selects adaptively the optimum one from a family set by the program user, according to the attenuation in the frequency bandwidth allocated to each MS. The frequency band allocated to each MS is also adaptively changed, according to the same attenuation.

The program displays dynamically, during the simulation, in its display window the number of error bits and error user-bins for each user and every SNR set by the program user as shown in the two figures below.



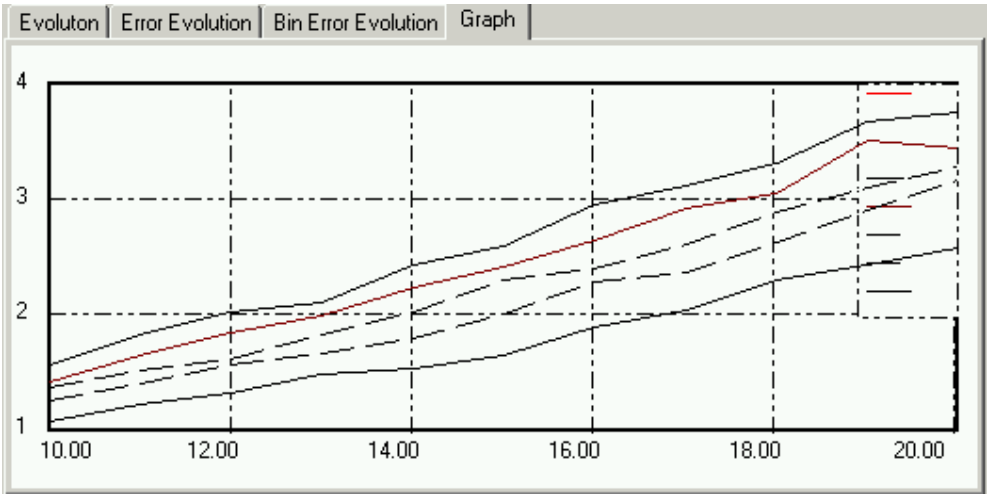
Evolution of the error bits vs. SNR for five users

Evoluton | Error Evolution | Bin Error Evolution | Graph

	10dB	11dB	12dB	13dB	14dB	15dB	16dB
User1	38	15	5	3	2	2	3
User2	39	18	7	7	9	5	1
User3	44	21	16	6	14	1	9
User4	59	28	18	9	11	14	0
User5	113	66	41	16	20	10	7

Evolution of the error bins vs. SNR for five users

After the simulation is over, the spectral efficiencies of each user and the global spectral efficiency vs. SNR curves are displayed, see the figure below.



Spectral efficiencies vs. SNR for each of the five MSs.

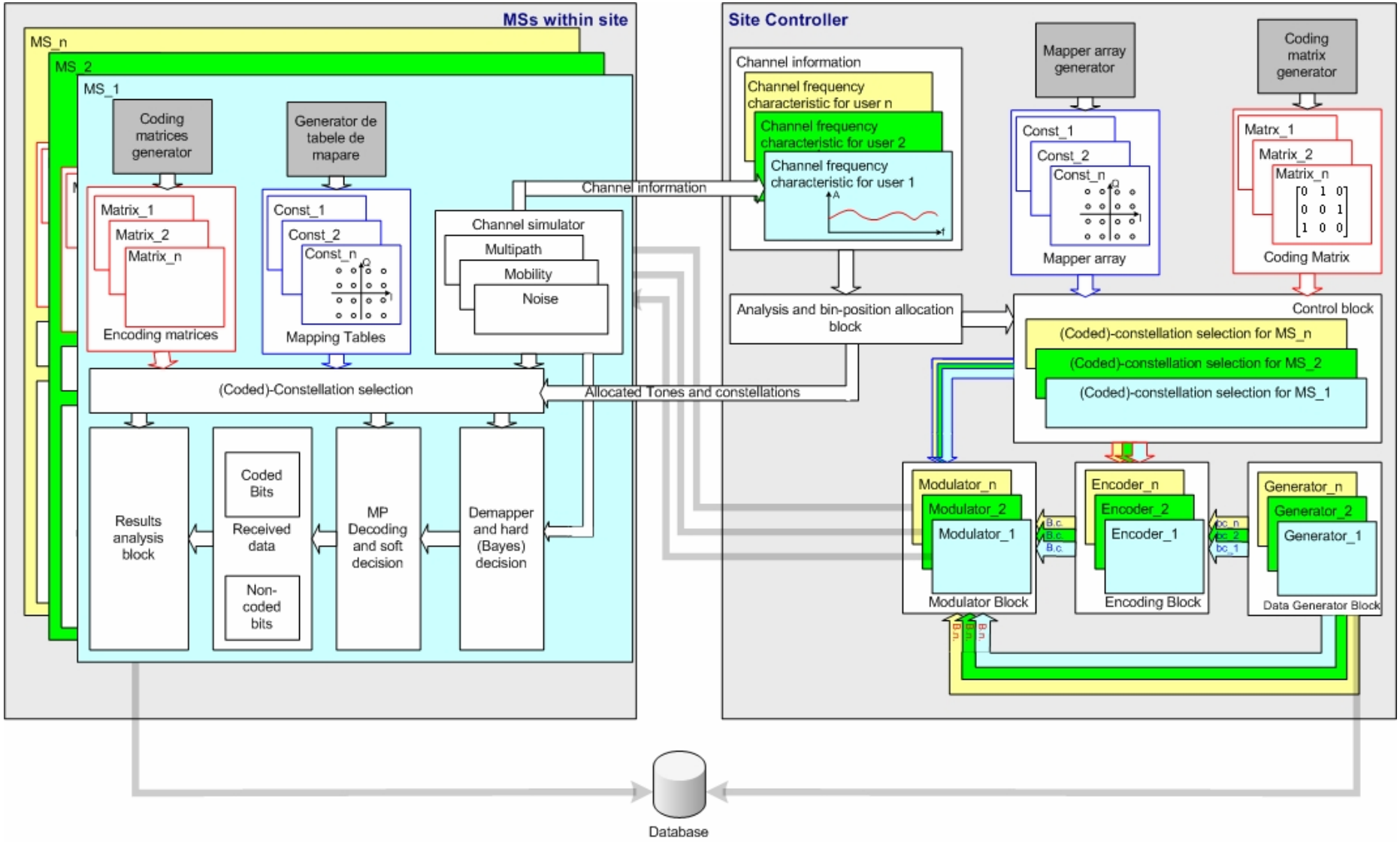


Figure 4. Block diagram of the program that simulates a site with multiple mobile stations