

# simARQ, An Automatic Repeat Request Simulator for Teaching Purposes

Carlos García-García, Juan Gómez-Luna, Ezequiel Herruzo Gomez,  
and José Ignacio Benavides Benitez

Computer Architecture, Electronics and Electronic Technology Dept. University of Córdoba,  
Spain

{i52gagac,e11goluj,e11hegoe,e11bebej}@uco.es

**Abstract.** Automatic Repeat Request (ARQ) protocols are a very important functionality in computer networks, which steer information transmission between network devices. Its knowledge is essential for students in Computer Science. In this way, simARQ simulator is an effective learning tool, which gives students an environment for self-training. It has Demo and Simulation modes, and a useful simulation analysis tool. simARQ has been successfully compared to other widely used ARQ simulator, through a survey among a group of students.

**Keywords:** Automatic Repeat Request, Sliding Window, Simulator, Teaching Tool.

## 1 Introduction

Nowadays, computer networks are present in every company or academic institution. They provide an easy and efficient way of communication and information exchange. In 1984, the International Organization of Standardisation [1] defined the Open Systems Interconnection (OSI) model, in order to simplify and organize communications between applications on different computers and different networks.

OSI model has 7 layers with different functionalities. The second layer, called data link layer, provides functions and procedures to transfer data between network entities and to detect errors that may occur in the physical medium. This functionality is implemented by Automatic Repeat reQuest (ARQ) protocols, such as *Stop-and-wait*, *Go-back-N* and *Selective-reject*. These protocols are also implemented in the Transport Layer.

In this way, a good understanding of this protocols and its operation is very important for students in Computer Science. Thus, it is essential to have a teaching application that effectively helps students in this purpose.

simARQ simulator has been developed at the University of Córdoba, in order to support teaching and give the students an environment to experiment. It implements Stop-and-wait, Go-back-N and Selective-reject protocols. In this way, simARQ helps students to understand these protocols. simARQ simulator is publicly available at [http://www.uco.es/~e11goluj/index\\_eng.html](http://www.uco.es/~e11goluj/index_eng.html).

Other tools with similar purposes have been found. Tanenbaum's simulator [2] is powerful but difficult to use, because it does not have a graphical user interface (GUI). The ARQ Simulator [3] is a simple animated simulator, but it does not permit to experiment with the protocols. Ingrase [4] is based on Tanenbaum's simulator. In fact, it is basically a GUI for that, with some improvements such as statistics.

In this work, simARQ has beaten Ingrase in a survey answered by a group of students.

The paper is organized as follows. Section 2 introduces the ARQ protocols implemented by simARQ. Section 3 describes simARQ simulator and its modules. A survey assessment of simARQ is detailed in Section 4. Finally, Section 5 concludes the paper.

## 2 A Quick Review on ARQ Protocols

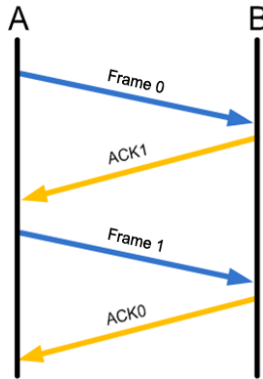
Unfortunately, communication lines make mistakes occasionally, due to attenuation, noise, propagation delays, etc. In order to solve these and ensure communications integrity, ARQ protocols were created.

ARQ protocols are error-control methods for data transmission that use *acknowledgements* (messages sent by the receiver indicating that it has correctly received a data frame or packet) and *timeouts* (specified periods of time allowed to elapse before an acknowledgment is to be received), in order to achieve reliable data transmission over an unreliable service. If the sender does not receive an acknowledgment (ACK) before the timeout, it usually re-transmits the frame until the sender receives an acknowledgment or exceeds a predefined number of re-transmissions.

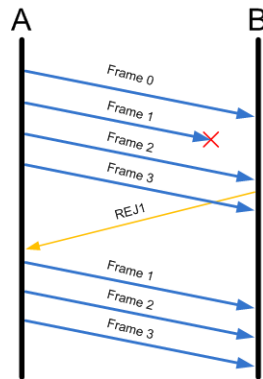
ARQ protocols are based on a flow control technique called *sliding window* [5]. When using this technique, a sender transmits some frames and requires the receiving device to send back an acknowledgment when it successfully has received the frames, which signals the sending device that another frame can be sent. *Window size* is the number of frames that can be sent without waiting for an ACK. With the sliding window method, the receiving device can send a single acknowledgement message for multiple frames sent in one window. Within that acknowledgement message is information about the receiving devices buffer size, which tells the sending device to increase or decrease the number of packets in the next transmission (this is where the "sliding" in the name comes in). If the application reading the data processes the frames at a slower rate than the sending device is transmitting them, it will tell the sending device to decrease the number of frames or temporarily cease transmission altogether, in order to free up room in the buffer; if the receiving application can process the frames faster than the sending device is transmitting them, it will tell the sending device to increase the number of packets in the next window as the applications buffer can handle more data.

There are three main ARQ protocols: Stop-and-wait, Go-back-N and Selective-reject. They are revised in the following lines.

A Stop-and-wait ARQ sender transmits one frame at a time; it is a special case of the general sliding window protocol with both transmit and receive window sizes equal to 1. After sending each frame, the sender does not transmit any further frames until it receives an ACK frame, as it is shown in Figure 1. After receiving a good frame, the receiver sends an ACK. If the ACK does not reach the sender before a certain time interval, known as timeout, the sender retransmits the same frame.



**Fig. 1.** Stop-and-wait performance: The sender A waits until receiving an acknowledgement (ACK), before sending the next frame



**Fig. 2.** Go-back-N performance: The sender A repeats transmissions from the frame that has failed

Go-Back-N ARQ is a protocol in which the sending process continues in order to send a number of frames specified by a window size, even without receiving an ACK, now called RR (Ready to Receive), from the receiver. Now, both windows (transmit and receive) will be different from zero. The window size is the maximum number of frames that can be sent without acknowledgement (RR). The receiver will only accept the next sequence number it is expecting. Other sequence numbers will be rejected (receiver will send a REJECT frame). Figure 2 illustrates Go-Back-N.

Selective-reject is similar to Go-back-N, since it also uses the sliding window technique. The main difference is that negative acknowledgments (REJ frames) are

selective (called SREJ) and the transmitter only retransmits frames that are in error, that is, the sender carries out sending a number of frames specified by the window size even after a frame loss. Unlike Go-back-N protocol, the receiver will continue accepting and sending acknowledges frames after an erroneous frame. This is shown in Figure 3.

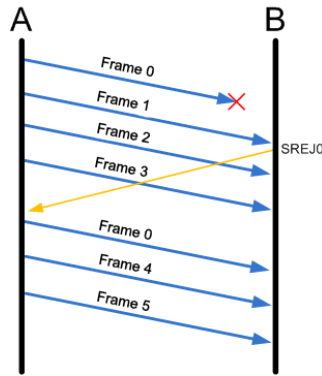


Fig. 3. Selective-reject performance: The sender A retransmits only the frame that has failed

### 3 simARQ Simulator

simARQ is an application for Microsoft Windows which implements the following ARQ protocols: Stop-and-wait, Go-back-N and Selective-reject. It has two main modes: Demo and Simulation. It also provides a simulation analysis tool, in order to thoroughly study simulation results.

The simulator has a friendly graphical user interface which allows the user to interact with the program easily. It has a menu bar in which users can select program options. It has a status bar, found at the bottom of windows, which reports the simulation state.

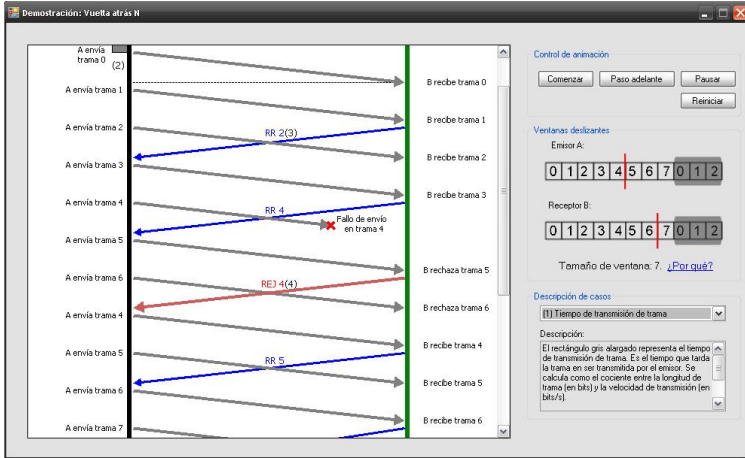
It also includes a complete help file, where the features and performance of the program are explained.

#### 3.1 Demo Mode

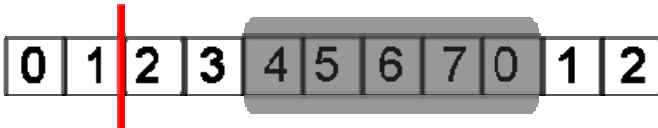
This mode aims to help students to understand how the protocols work. It contains a chart, which shows step by step the performance of each protocol. This is accompanied by a help text that explains each step of the protocol.

Demos show all possible cases of the protocols. That is, any possible condition is shown and explained. Demos have the following features:

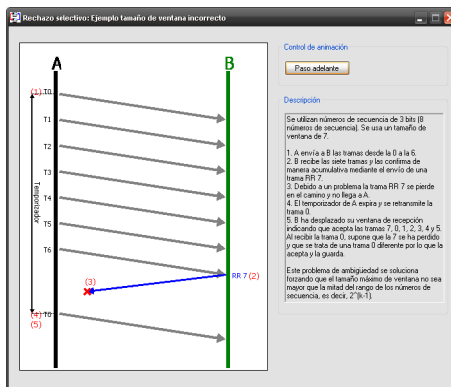
- Protocol to be shown: Stop-and-wait, Go-back-N or Selective-reject.
- Animated chart, in which the process is plotted step by step. As Figure 4 shows, the sending host A lies on the left and the receiving host B on the right. The chart represents the transmission times of the different frames and the propagation delays.
- Chart animation controls: start, step-by-step, pause, restart.



**Fig. 4.** Demo mode. On the left, the animated chart shows the performance of the protocol. On the right, animation controls, state of the sliding windows and dropdown list with cases explanation.



**Fig. 5.** Sliding window. Sequential numbers represent frames that are being sent; semitransparent area stands for current position of the window; red line marks which are the frames that have been confirmed.



**Fig. 6.** Maximum window size. Ambiguous cases are shown, in order to explain why the window size should be limited.

- Cases are thoroughly explained in a text box beside the animated chart. There are several events on the animated chart, for which the user can use the dropdown list to query the corresponding explanation.

- State of the sliding window: if the protocol uses the sliding window technique (Go-back-N and Selective-reject), the status of both windows (sender and receiver) will be displayed at all times. Figure 5 explains the sliding window representation.
- Explanation about maximum window sizes: protocols using the sliding window technique have a limitation in the window size, in order to avoid any ambiguity while interpreting the sequence of frames [5]. Figure 6 shows an ambiguous case for Selective-reject.

### 3.2 Simulation Mode

The simulation mode generates simulations of the three ARQ protocols. The following functions are implemented:

- Protocol to be simulated: Stop-and-wait, Go-back-N or Selective-reject.
- Simulation parameters are introduced through a form. These parameters can later be easily modified, in order to carry out several simulations through one parameter range.
- Simulations can be loaded from/saved to an external file.
- Once the simulation has been performed, statistical graphics about correct and wrong frames are generated.

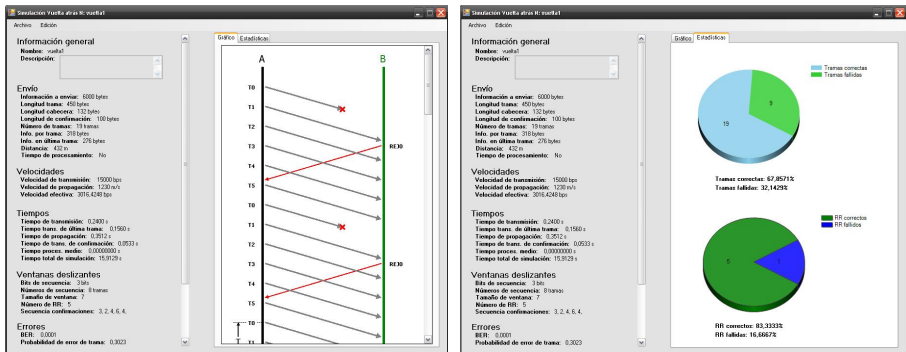
The screenshot shows a window titled "Nueva simulación: Rechazo selectivo". It contains several input fields and buttons. On the left, there are fields for "Nombre:", "Descripción:", and a section titled "Parámetros" with fields for "Tamaño total información (bytes)", "Longitud de trama (bytes)", "Longitud de cabecera (bytes)", "Longitud de RR (bytes)", "Velocidad de transmisión (bps)", "Velocidad propagación (m/s)", "Distancia (m)", "Intervalo de expiración (s)", and "Probabilidad error de bit (BER)". On the right, there are buttons for "Simular", "Cancelar", and "Abrir...". Below these are sections for "Ventanas deslizantes" with radio buttons for "3 bits" (selected) and "7 bits", and a "Tamaño de ventana:" field. At the bottom right, there is a "Tiempo procesamiento" section with radio buttons for "Sí" and "No" (selected).

**Fig. 7.** Simulation parameters form permits to introduce simulation details. Then, by clicking on “Simulate”, the simulation is started.

Simulation parameters form is shown in Figure 7. These parameters are: information size, frame length, header length, and RR/ACK length, in bytes; transmission rate, in bits per second (bps); propagation speed, in meters/second (m/s); distance, in meters (m); timeout, and processing time in host devices, in seconds (s); bit error rate (BER) of the physical communication line; and bits used for sequence numbers.

Once the parameters have been inserted and “Simulate” button has been pushed, the system checks if all the parameters have correct values. If a parameter is incorrect, it returns a warning message. If all the parameters are correct, the application starts the simulation.

When the simulation finishes, the results window pops-up. This window, shown in Figure 8, presents many simulation results on its left side: number of frame transmissions and retransmissions; transmission time and propagation time (s) of frame and RR/ACK; total simulation time (s); effective rate (bps), as the ratio between the number of bytes received and the total simulation time; number of wrong frames and RR/ACKs; frame and RR/ACK error probabilities; etc. On the right side, there are two tabs: “Chart” and “Statistics”. The former shows the complete communication process along time, indicating transmissions, acknowledgements, rejects, timeouts and retransmissions. The latter shows statistical results about frames and RR/ACKs.

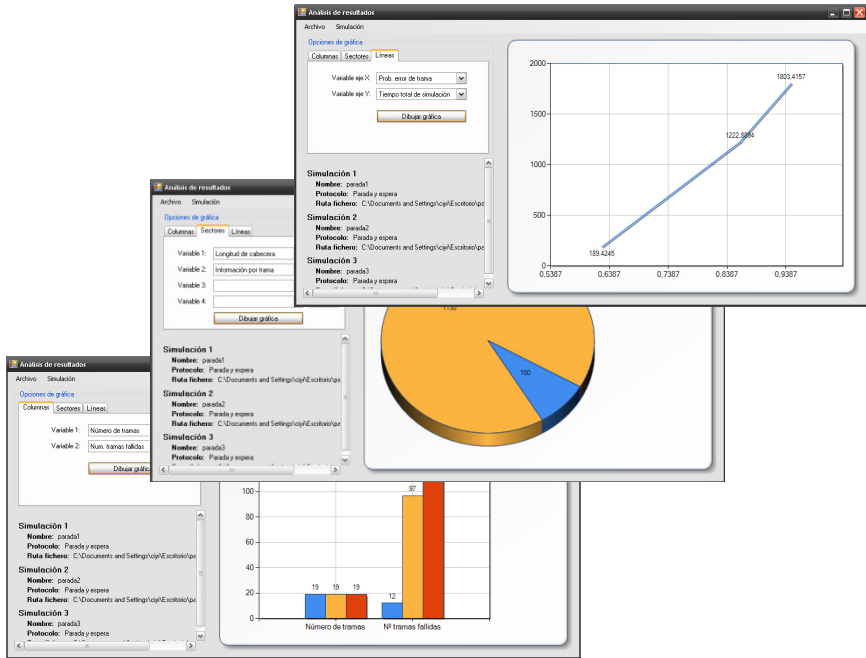


**Fig. 8.** Simulation results. On the left of the window, simulation results are shown. By clicking on “Chart” tab (left), the simulation chart is shown. By clicking on “Statistics” tab (right), statistics about wrong and correct frames are displayed.

### 3.3 Simulation Analysis Tool

This tool permits users to represent the results of one or several simulations, in order to analyze the performance of ARQ protocols. It boosts the usefulness of the simulator, since protocols may be exhaustively studied, changing the simulation parameters and comparing the subsequent simulation results. None of the previous ARQ simulators [2][3][4] offer such an analysis utility.

Each of the magnitudes that result from the simulations may be displayed on column, pie or line charts. Users can save the plots as bitmap files. Display controls are very flexible, allowing users to represent several magnitudes of one simulation or one magnitude of several simulations. For instance, Figure 9 (left) depicts the number of wrong frames in three Stop-and-wait simulations. The three transmit the same amount of information in identical transmission conditions (transmission rate, timeout...), but the BER of the communication line takes three different values. In Figure 9 (middle), a pie chart shows the header length and the information field length used in one simulation. Figure 9 (right) presents a line chart showing the dependence of the total simulation time on the frame error probability.



**Fig. 9.** Simulation analysis tool. On the left, a column chart for three simulations. In the middle, a pie chart for header and information length of the frames of one simulation. On the right, a line chart represents the total simulation time with respect to the frame error probability.

## 4 simARQ Assessment

In order to assess simARQ, during the 2010 academic year, a 3-hour laboratory session about ARQ protocols was proposed to a group of 20 volunteer students who attended to the course “Computer Networks” at the University of Córdoba. They also worked with Ingrase simulator. At the end of the laboratory session, they were asked to complete a questionnaire, in order to know their subjective perceptions about both simulators.

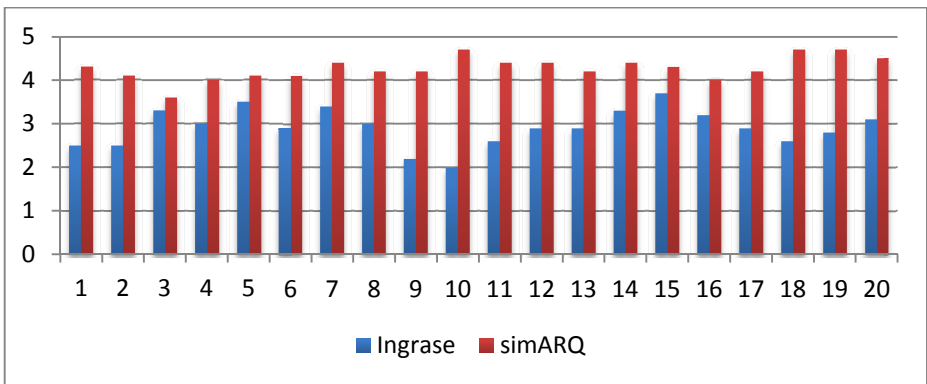
The questionnaire was filled out through the Web Evaluation Teaching Tool (WETT) [6]. This is a web-based survey platform, which permits instructors to prepare specific questionnaires about learning tools. Students answer on a 5-point scale, where 5 and 0 indicate strong acceptance or rejection, respectively. Later, the WETT reports the average score and the standard deviation of each question, a group of questions related to the same topic, and the global score of the tool.

The survey contained twenty questions, which are shown in Table 1. Questions evaluated the goodness of the user interface, the features of the simulator, the correspondence between the simulator and theoretical concepts, and how the tool can be used for self-training.



**Table 1.** Questionnaire proposed to students, after working with Ingrase and simARQ

Questions
1. The simulator is easy-to-use and intuitive
2. The interface is pleasant and clear
3. The toolbar is useful
4. It is easy to insert the parameters for simulations
5. It is possible to save the simulation results and to organize these files
6. It is easy to understand the meaning of chart symbols and results after a simulation
7. Results charts (text files in Ingrase) are clear and intelligible
8. The simulator offers enough information about each magnitude or parameter
9. The help files are useful, comprehensive and easy-to-use
10. The Demo mode clearly explains how the protocols work
11. The simulator makes easier understanding theoretical concepts related to ARQ protocols
12. Performance of ARQ protocols is easier to understand thanks to the simulator
13. Rare or special cases are easier to understand with the simulator
14. Simulation results are complete, clear and easily understandable
15. Statistical results are clear and useful
16. The results analysis is easy with this simulator
17. ARQ protocols and different transmission scenarios may be clearly compared within the simulator
18. From a general point of view, this simulator is a good teaching tool.
19. From a general point of view, this simulator helps the student with the contents of the course
20. From a general point of view, this simulator will improve the quality of teaching.



**Fig. 10.** Survey results for Ingrase and simARQ. Questions 1 to 20 are shown in Table 1

Results in Figure 10 are encouraging, since simARQ beats Ingrase in all the questions. simARQ scores at least 3.6 in every question. Moreover, simARQ has a global score of 4.28, while Ingrase has 2.91.

## 5 Conclusions

simARQ simulator presented in this paper is a very useful tool for teaching purposes, in order to enhance students' knowledge about ARQ protocols. It has a Demo mode and a Simulation mode. Moreover, it has simulation analysis tool, which allows users to analyze the simulation results.

simARQ is used in laboratory practices of the course "Computer Networks" at the University of Córdoba. It has been compared with the previously used simulator, Ingrase, through a survey to students. This has revealed that students consider simARQ a much better learning tool than Ingrase.

## References

1. International Organization of Standardization, <http://www.iso.org/>
2. Tanenbaum, A.S.: Computer Networks, 4th edn. Pearson (2003)
3. Akif, B.T., Mehmet, F.I., Mustafa, A., Sinan, K., Ahmet, B., Tuba, D.: Arq Simulator, [http://www.ee.hacettepe.edu.tr/~aktas/arq\\_simulator/](http://www.ee.hacettepe.edu.tr/~aktas/arq_simulator/)
4. University of Seville. Ingrase Simulator, <http://trajano.us.es/~fornes/ARSSP/>
5. Stallings, W.: Data and Computer Communications, 7th edn. Pearson (2003)
6. Gómez-Luna, J., Herruzo, E., Benavides, J.I.: WETT: Evaluating Teaching Tools and Methodologies. In: International Conference on Computational Methods in Sciences and Engineering 2008. AIP Conference Proceedings, vol. 1148, pp. 527–530 (2008)