4 de septiembre de 2023.
Dictamen C.I. 14/2023

DICTAMEN
QUE PRESENTA LA COMISIÓN DE INVESTIGACIÓN DE LA DIVISIÓN DE CIENCIAS DE LA COMUNICACIÓN Y DISEÑO

ANTECEDENTES

I. El Consejo Divisional de Ciencias de la Comunicación y Diseño, en la sesión 08.23, celebrada el 2 de mayo de 2023, integró esta Comisión en los términos señalados en el artículo 55 de Reglamento Interno de los Órganos Colegiados Académicos.

II. El Consejo Divisional designó para esta Comisión a los siguientes integrantes:

a) Órganos personales:
- ✓ Dra. Margarita Espinosa Meneses, Jefa del Departamento de Ciencias de la Comunicación;
- ✓ Dra. Erika Cecilia Castañeda Arredondo, Jefa del Departamento de Teoría y Procesos del Diseño;
- ✓ Dr. Carlos Roberto Jaimez González, Jefe del Departamento de Tecnologías de la Información.

b) Representantes propietarios:
- Personal académico:
  - ✓ Dr. Diego Carlos Méndez Granados, Departamento de Ciencias de la Comunicación;
  - ✓ Dr. Manuel Rodríguez Viqueira, Departamento de Teoría y Procesos del Diseño;
  - ✓ Mtra. Betzabet García Mendoza, Departamento de Tecnologías de la Información.

CONSIDERACIONES

I. La Comisión recibió, para análisis y discusión, el reporte de resultados y la solicitud de cierre del proyecto de investigación denominado **"Almacenamiento seguro de información en redes P2P usando técnicas de codificación de red y dispersión de información"** presentado por el Dr. Francisco de Asís López Fuentes, aprobado en la Sesión 03.20 celebrada el 27 de febrero de 2020, mediante el acuerdo DCCD.CD.10.03.20.

II. En sesión 04.22 celebrada el 2 de febrero de 2022 se aprobó el primer reporte parcial de resultados del proyecto de investigación.

III. La Comisión de Investigación sesionó el 4 de septiembre de 2023, fecha en la que concluyó su trabajo de análisis y evaluación del reporte de resultados, con el presente Dictamen.

IV. La Comisión tomó en consideración los siguientes elementos:

- *"Lineamientos para la creación de grupos de investigación y la presentación, seguimiento y evaluación de proyectos de investigación"* aprobados en la Sesión 06.16 del Consejo Divisional de Ciencias de la Comunicación y Diseño, celebrada el 6 de junio de 2016, mediante al acuerdo DCCD.CD.15.06.16.
- Protocolo de investigación.
- Relevancia para el Departamento.
- Objetivos planteados.
- Resultados obtenidos.

V. **Objetivo general**:

Estudiar cómo técnicas de codificación de red y dispersión de información asociadas a mecanismos de seguridad impacta en el almacenamiento distribuido en redes peer-to-peer.

VI. **Logros alcanzados**:

Comprometido	Realizado
Publicación de 3 artículos de investigación (2 artículos en congresos internacionales y un artículo en revista) Escritura de un reporte técnico para futura publicación como artículo de investigación.	Publicación de 6 artículos de investigación: - 1 artículo presentado en congreso internacional IEEE - 4 artículos presentados en congresos internacionales publicados por Springer (indizados en Scopus) - 1 artículo presentado en congreso internacional en proceso de publicación por Springer (Scopus).

	1. López-Fuentes, F. A., Ortega-Vallejo R. A. y Marcelín-Jiménez R. (2021). "Network Coding and Dispersal Information with TCP for Content Delivery", *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications* Engineering book series (LNICST, volume 393), Springer-Nature. Scopus. https://doi.org/10.1007/978-3-030-87495-7_5 Artículo presentado en la *5th EAI International Conference on Computer Science and Engineering in Health Services* (29 y 30 de julio de 2021). Artículo obtuvo *"Best Paper Award"*. 2. López-Fuentes F.A. (2021). "Content Distribution and Storage Based on Volunteer and Community Computing". In: Mata-Rivera M.F., Zagal-Flores R. (eds) Telematics and Computing. WITCOM 2021. Communications in Computer and Information Science, vol 1430. Springer, Cham. Scopus. https://doi.org/10.1007/978-3-030-89586-0_13 Artículo presentado en el *International Congress in Telematics and Computing* (12 y 13 de noviembre de 2021). 3. Ortega Vallejo, R. A. and López-Fuentes, F. A. (2022). "Obfuscation in Network Coding to Mitigate the Effects of Pollution Attacks". In: Mata-Rivera, M. F., Zagal-Flores, R. and Barria-Huidobro, C. (eds).Telematics and Computing. WITCOM 2022. Communications in Computer and Information Science, vol. 1659. Springer, Cham. Scopus. https://link.springer.com/chapter/10.1007/978-3-031-18082-8_1 Artículo presentado en la *11th International Conference WITCOM 2022* (9 de noviembre de 2022). 4. López-Pedroza I. A., López-Fuentes F. A. (2021) "Teaching of TCP Fundamental Operations using a

	Digital Tool". In: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering book series (LNICST, volume 359), Springer-Nature. Scopus. https://link.springer.com/chapter/10.1007/978-3-030-69839-3_16 Artículo presentado en la *4th EAI International Conference on Computer Science and Engineering (COMPSE 2021).* 5. López-Fuentes, F.A. (2022). Didactic Tool for Teaching Quality of Service Algorithms in Communication Networks, 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON 2022). https://ieeexplore.ieee.org/document/9946543 Artículo presentado en la IEMCON 2022. 6. Artículo en proceso de publicación: "An Information Architecture for the Engineering and Design of Industrial Electrical Systems". Artículo presentado en la 6th EAI International Conferen
Software de simulación resultante del proyecto	Se crearon varios programas (software) para simular diferentes condiciones en redes de computadoras. Estos programas se probaron remotamente en un clúster de computadoras de la UAM Iztapalapa.
	Actividades adicionales: miembro del comité técnico y revisor de varios congresos IEEE. - Miembro del Comité Técnico de la "12th Annual IEEE Information Technology, Electronics and Mobile Communication Conference". 2021. - Miembro del Comité Técnico de la "13th Annual IEEE Information technology, Electronics and Mobile Communication Conference". 2022. - Miembro del Comité Revisor de la "IEEE UEMCON 2021". - Miembro del Comité Revisor del "Journal of Intelligent & Fuzzy Systems". 2021.

	- Miembro del Comité Revisor de la "IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)".

DICTAMEN

ÚNICO:
Tras evaluar el reporte de resultados y la solicitud de cierre del proyecto de investigación denominado **"Almacenamiento seguro de información en redes P2P usando técnicas de codificación de red y dispersión de información"** presentado por el Dr. Francisco de Asís López Fuentes, la Comisión de Investigación recomienda al Consejo Divisional de Ciencias de la Comunicación y Diseño aceptarlo.

Con el reporte presentado se cierra el proyecto **"Almacenamiento seguro de información en redes P2P usando técnicas de codificación de red y dispersión de información"**.

VOTOS:

Integrantes	**Sentido de los votos**
Dra. Margarita Espinosa Meneses	A favor
Dra. Erika Cecilia Castañeda Arredondo	A favor
Dr. Carlos Roberto Jaimez González	A favor
Dr. Diego Carlos Méndez Granados	------
Dr. Manuel Rodríguez Viqueira	A favor
Mtra. Betzabet García Mendoza	A favor
Total de los votos	**5 votos a favor**

Coordinadora

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
UNIDAD CUAJIMALPA
SECRETARÍA ACADÉMICA
DIVISIÓN DE CIENCIAS DE LA
COMUNICACIÓN Y DISEÑO

Mtra. Si[redacted]ínez
Secretaria del Consejo Divisional de
Ciencias de la Comunicación y Diseño

DCCD.DTI.062.23
Ciudad de México, 31 de agosto de 2023

Dra. Gloria Angélica Martínez de la Peña
Presidenta del Consejo Divisional de la
División de Ciencias de la Comunicación y Diseño
Universidad Autónoma Metropolitana, Unidad Cuajimalpa
P R E S E N T E

Asunto: Entrega de Informe Final de
Proyecto de Investigación

Con relación al asunto arriba referido, por este conducto me permito solicitarle sea turnado a la Comisión de Investigación y al Consejo Divisional, para su revisión, y en su caso aceptación, el informe final de trabajo correspondiente al Proyecto de Investigación titulado ***Almacenamiento seguro de información en redes P2P usando técnicas de codificación de red y dispersión de información***, cuyo responsable es el **Dr. Francisco de Asís López Fuentes.** Este proyecto de investigación fue aprobado en la Sesión 03.20 del Consejo Divisional de Ciencias de la Comunicación y Diseño, celebrada el 27 de febrero de 2020, mediante el **acuerdo DCCD.CD.10.03.20**.

Se anexa cuadro comparativo de los productos comprometidos y los productos realizados, así como la carta de finalización y el informe final del proyecto de investigación.

Sin otro particular, le envío un cordial saludo.

A t e n t a m e n t e
Casa abierta al tiempo

Dr. Carlos Roberto Jaimez González
Jefe del Departamento de Tecnologías de la Información
UAM Cuajimalpa

c.c.p.: Mtra. Silvia Gabriela García Martínez – Secretaria del Consejo Divisional
Lic. Inés Andrea Zepeda Martínez – Oficina Técnica del Consejo Divisional

Comprometido	Realizado
Publicación de 3 artículos de investigación (2 artículos en congresos internacionales y un artículo en revista) Escritura de un reporte técnico para futura publicación como artículo de investigación.	Publicación de 6 artículos de investigación: - 1 artículo presentado en congreso internacional IEEE - 4 artículos presentados en congresos internacionales publicados por Springer (indizados en Scopus) - 1 artículo presentado en congreso internacional en proceso de publicación por Springer (Scopus). 1. López-Fuentes, F. A., Ortega-Vallejo R. A. y Marcelín-Jiménez R. (2021). “Network Coding and Dispersal Information with TCP for Content Delivery”, *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications* Engineering book series (LNICST, volume 393), Springer-Nature. Scopus. https://doi.org/10.1007/978-3-030-87495-7_5 Artículo presentado en la *5th EAI International Conference on Computer Science and Engineering in Health Services* (29 y 30 de julio de 2021). Artículo obtuvo *“Best Paper Award”.* 2. López-Fuentes F.A. (2021). “Content Distribution and Storage Based on Volunteer and Community Computing”. In: Mata-Rivera M.F., Zagal-Flores R. (eds) Telematics and Computing. WITCOM 2021. Communications in Computer and Information Science, vol 1430. Springer, Cham. Scopus. https://doi.org/10.1007/978-3-030-89586-0_13 Artículo presentado en el *International Congress in Telematics and Computing* (12 y 13 de noviembre de 2021). 3. Ortega Vallejo, R. A. and López-Fuentes, F. A. (2022). “Obfuscation in Network Coding to Mitigate the Effects of Pollution Attacks”. In: Mata-Rivera, M. F., Zagal-Flores, R. and Barria-Huidobro, C. (eds).Telematics and Computing. WITCOM 2022. Communications in Computer and Information Science, vol. 1659. Springer, Cham. Scopus. https://link.springer.com/chapter/10.1007/978-3-031-18082-8_1 Artículo presentado en la *11th International Conference WITCOM 2022* (9 de noviembre de 2022). 4. López-Pedroza I. A., López-Fuentes F. A. (2021) “Teaching of TCP Fundamental Operations using a Digital Tool”. In: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering book series (LNICST, volume 359), Springer-Nature. Scopus. https://link.springer.com/chapter/10.1007/978-3-030-69839-3_16 Artículo presentado en la *4th EAI International Conference on Computer Science and Engineering (COMPSE 2021).*

	5. López-Fuentes, F.A. (2022). Didactic Tool for Teaching Quality of Service Algorithms in Communication Networks, 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON 2022). https://ieeexplore.ieee.org/document/9946543 Artículo presentado en la IEMCON 2022. 6. Artículo en proceso de publicación: "An Information Architecture for the Engineering and Design of Industrial Electrical Systems". Artículo presentado en la 6th EAI International Conference on Computer Science and Engineering (COMPSE 2022).
Software de simulación resultante del proyecto	Se crearon varios programas (software) para simular diferentes condiciones en redes de computadoras. Estos programas se probaron remotamente en un clúster de computadoras de la UAM Iztapalapa.
	Actividades adicionales: miembro del comité técnico y revisor de varios congresos IEEE. - Miembro del Comité Técnico de la "12th Annual IEEE Information Technology, Electronics and Mobile Communication Conference". 2021. - Miembro del Comité Técnico de la "13th Annual IEEE Information technology, Electronics and Mobile Communication Conference". 2022. - Miembro del Comité Revisor de la "IEEE UEMCON 2021". - Miembro del Comité Revisor del "Journal of Intelligent & Fuzzy Systems". 2021. - Miembro del Comité Revisor de la "IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)".

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

Comunidad académica comprometida con el desarrollo humano de la sociedad.

Ciudad de México, mayo 30 de 2023

Dr. Carlos Roberto Jaimez González
Jefe del Departamento de Tecnologías de la Información
UAM Cuajimalpa
Presente

Por este conducto me permito entregar el Informe Final del proyecto intitulado "**ALMACENAMIENTO SEGURO DE INFORMACIÓN EN REDES P2P USANDO TÉCNICAS DE CODIFICACIÓN DE RED Y DISPERSIÓN DE INFORMACIÓN"** que se aprobó con el acuerdo DCCD.CD.10.03.20 de la sesión 03.20 llevada a cabo el 27 de febrero de 2020.

El informe presenta una relación de los productos obtenidos al término del proyecto, entre los resultados destacados se encuentran:

1 servicio social concluido
2 proyecto terminal concluido
1 artículos en conferencia internacional IEEE
4 artículos en conferencias internacionales publicadas por Springer (indexadas en Scopus)
1 artículo en conferencia internacional de EAI en proceso de publicación por Springer (Scopus)
Comité técnico y revisor de 19 artículos de investigación en conferencias IEEE.

Le agradezco de antemano presente este informe de terminación de proyecto de investigación ante el Consejo Divisional de la División de Ciencias de la Comunicación y Diseño.

Sin más por el momento, le agradezco la atención a la presente.

Atentamente

Casa abierta al tiempo

Dr. Francisco de Asís López Fuentes
Profesor-investigador Responsable del Proyecto de Investigación
Departamento de Tecnologías de la Información
Correo electrónico: flopez@cua.uam.mx

INFORME FINAL DE PROYECTO DE INVESTIGACIÓN

NOMBRE DEL PROYECTO: "ALMACENAMIENTO SEGURO DE INFORMACIÓN EN REDES P2P USANDO TÉCNICAS DE CODIFICACIÓN DE RED Y DISPERSIÓN DE INFORMACIÓN"

Fecha del reporte: 30 de mayo de 2023
Fecha de aprobación del proyecto: 27 de febrero de 2020
Período reportado: 28 de febrero de 2020 al 27 de febrero de 2023

El proyecto **"Almacenamiento seguro de información en redes P2P usando técnicas de codificación de red y dispersión de información"** fue aprobado por el Consejo Divisional de la DCCD en el acuerdo DCCD.CD.10.03.20 con fecha del 27 de febrero de 2020 y tiene como objetivo general el siguiente:

Estudiar cómo técnicas de codificación de red y dispersión de información asociadas a mecanismos de seguridad impacta en el almacenamiento distribuido en redes peer-to-peer.

Los participantes de este proyecto son:

- Dr. Francisco de Asís López Fuentes (responsable del proyecto)
- Dr. Ricardo Marcelín Jiménez (RMJ) – UAM Iztapalapa

I. RESULTADOS:

Los siguientes resultados fueron alcanzados:

A. FORMACIÓN DE RECURSOS HUMANOS:

El proyecto permitió involucrar a alumnos de licenciatura en Tecnologías y Sistemas de Información.

- **Alumnos en proyecto terminal:**

llse Alicia López Pedroza – Concluyo su proyecto terminal en el trimestre 20-I

Hossein Yahyazadeh – inició su proyecto terminal en el trimestre 21-O y concluyó su proyecto terminal en 22P (25 septiembre 2022).

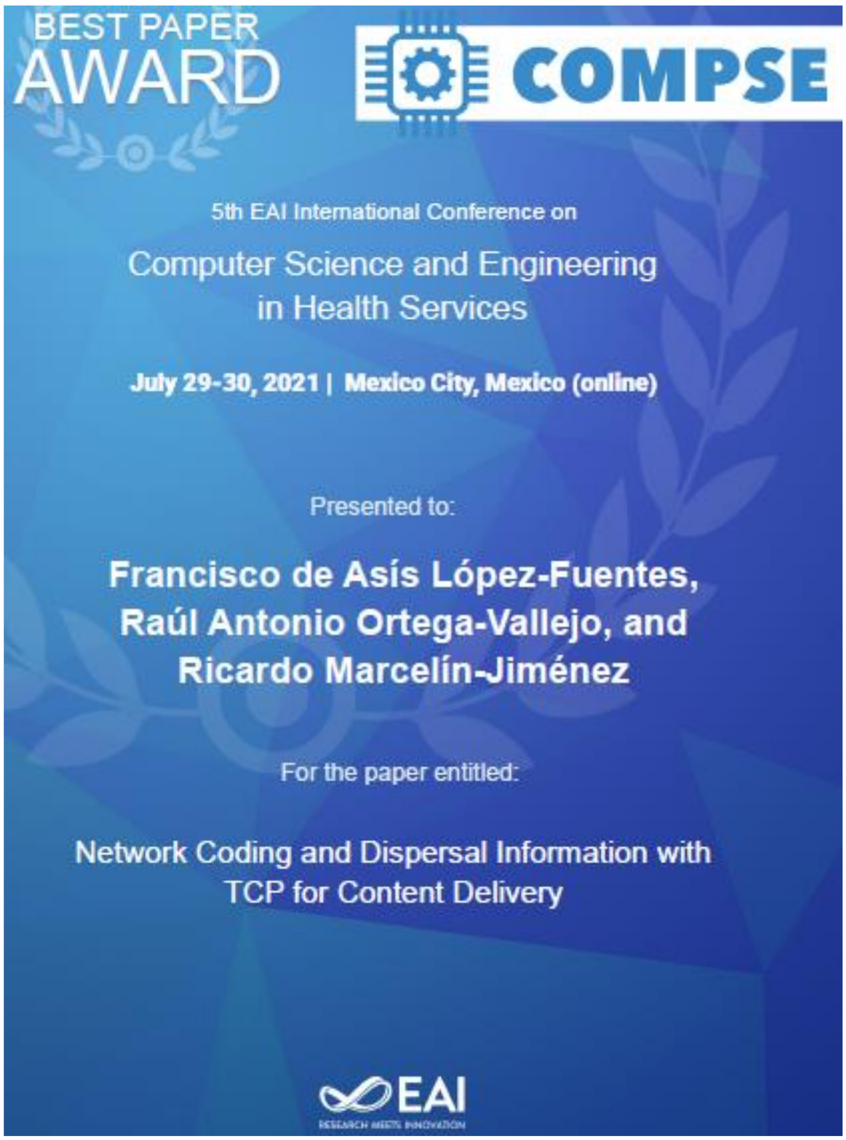
- **Alumnos en servicio social:**

Raúl Antonio Ortega Vallejo – realizó y concluyó su servicio social entre del 21 de Julio del 2020 al 27 de abril de 2021.

B. PRESENTACIÓN DE RESULTADOS EN EVENTOS INTERNACIONALES

Los resultados obtenidos de este proyecto se presentaron en los siguientes eventos:

1. En la 5th EAI International Conference on Computer Science and Engineering in Health Services, 29 y 30 de julio de 2021 (realizado en modo virtual) se present el artículo Network coding and dispersal information with TCP for content delivery". En esta conferencia se otorgó a nuestro trabajo "el "Best Paper Award" del evento. Se adjunta constancia.



2. En The International Congress in Telematics and Computing (realizado en modo virtual los días 12 y 13 de noviembre de 2021) se present el artículo Content distribution and storage based on Volunteer and Community Computing.

3. En la 11th International Conference WITCOM 202 se presentó el artículo "Data Obfuscation in Network Coding to Mitigate the effects of pollution Attacks" (presentada en modo virtual el 9 de noviembre de 2022)

4. Durante el desarrollo de este proyecto también se realizó la presentación del trabajo "Teaching of TCP Fundamentals Operation using a Digital Tool" en la 4Th EAI International Conference on Computer Science and Engineering (COMPSE 2020).

COMPSE

EAI
RESEARCH MEETS INNOVATION

Certificate of Participation

4th EAI International Conference on Computer Science and Engineering in Health Services (EAI COMPSE 2020)

November 26th, 2020
held virtually

This certificate confirms that **López-Fuentes, Francisco De Asís**

presented the paper ***Teaching Of Tcp Fundamental Operations Using A Digital Tool***

at the conference EAI COMPSE 2020

5. Durante el desarrollo del proyecto también se realizó la presentación del trabajo "An Information Architecture for the Engineering and Design of Industrial Electrical Systems" en la 6Th EAI International Conference on Computer Science and Engineering (COMPSE 2022).

COMPSE

EAI
RESEARCH MEETS INNOVATION

Certificate of Participation

6th EAI International Conference on Computer Science and Engineering (COMPSE 2022)

October 6, 2022 Mexico City, Mexico

This certificate confirms that

Francisco de Asís López-Fuentes

participated at the conference with their research paper titled:

An Information Architecture for the Engineering and Design of Industrial Electrical Systems

Veronika Kissova
EAI Conference Manager

6. Durante el desarrollo del proyecto también se realizó la presentación del trabajo "Didactic tool for teaching quality of service algorithms in communication networks" en IEEE IEMCON 2022 (13th IEEE Information Technology, Electronics and Mobile Communication Conference.

Las asistencias virtuales a estas conferencias fueron soportada completamente por el Departamento de Tecnologías de la Información - UAM Cuajimalpa.

C. PUBLICACIONES

Como resultado de este proyecto se realizaron las siguientes publicaciones:

1. **López-Fuentes, F. A.,** Ortega-vallejo R. A. y Marcelín-Jimenez R. “Network Coding and Dispersal Information with TCP for Content Delivery” Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering book series (LNICST, volume 393), Springer-Nature, 2021.

Springer Link

International Conference on Computer Science and Health Engineering

↳ COMPSE 2021: **Computer Science and Engineering in Health Services** pp 63–72 | Cite as

Home › Computer Science and Engineering in Health Services › Conference paper

Network Coding and Dispersal Information with TCP for Content Delivery

Francisco de Asís López-Fuentes, Raúl Antonio Ortega-Vallejo & Ricardo Marcelín-Jiménez

Conference paper | First Online: 29 September 2021

Part of the Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering book series (LNICST,volume 393)

Abstract

Dissemination information from many sources to many receivers can be fundamental in different systems. However, the components of these system may present some failure type both the software and hardware. In addition, problems related to the communication networks such as limited bandwidth or packet loss should be present. The information dispersal algorithm (IDA) has been used as a good solution to offer fault tolerance. On the other hand, network coding is a coding method mainly used to increase throughput of a communication

Enlace: https://doi.org/10.1007/978-3-030-87495-7_5

2. **López-Fuentes F.A. (2021)** Content Distribution and Storage Based on Volunteer and Community Computing. In: Mata-Rivera M.F., Zagal-Flores R. (eds) Telematics and Computing. WITCOM 2021. Communications in Computer and Information Science, vol 1430. Springer, Cham.

SpringerLink

International Congress of Telematics and Computing

WITCOM 2021: Telematics and Computing pp 163-173 | Cite as

Content Distribution and Storage Based on Volunteer and Community Computing

Authors   Authors and affiliations

Francisco de Asís López-Fuentes

Conference paper
First Online: 01 November 2021

Part of the Communications in Computer and Information Science book series (CCIS, volume 1430)

Abstract

In recent years, the users need to discover and use a diversity of resources in the Internet to do their tasks. These resources such as massive storage, processing and distribution capacity are generally decentralized and geographically dispersed, however they can be shared to solve large-scale problems in a collaborative way. Peer-to-peer (P2P) networks are an attractive alternative to implement collaborative solutions. This work presents a P2P collaborative for content distribution and store management in small communities based on volunteer and community computing.

Enlace: https://doi.org/10.1007/978-3-030-89586-0_13

3. Ortega Vallejo, R. A. and **López-Fuentes**, F. A. (2021). “Obfuscation in Network Coding to Mitigate the Effects of Pollution Attacks”. In: Mata-Rivera, M. F., Zagal-Flores, R. and Barria-Huidobro, C. (eds).Telematics and Computing. WITCOM 2022. Communications in Computer and Information Science, vol. 1659. Springer, Cham

Springer Link

International Congress of Telematics and Computing

↳ WITCOM 2022: **Telematics and Computing** pp 1–17 | Cite as

Home > Telematics and Computing > Conference paper

Data Obfuscation in Network Coding to Mitigate the Effects of Pollution Attacks

Raúl Antonio Ortega-Vallejo & Francisco de Asís López-Fuentes

Conference paper | First Online: 30 October 2022

357 Accesses

Part of the Communications in Computer and Information Science book series (CCIS,volume 1659)

Abstract

Network coding is a technique mainly used to maximize the throughput, minimize the delay, or optimize the reliability in the communication networks. However, network coding presents vulnerabilities problems in security terms and is susceptible to security attacks. We analyze impact of a security attack called pollution attack in traditional network coding based on butterfly scheme and propose a solution to deal with this problem. Cryptographic algorithms

Enlace: https://link.springer.com/chapter/10.1007/978-3-031-18082-8_1

4. López-Pedroza I. A., **López-Fuentes F. A.** "Teaching of TCP Fundamental Operations using a Digital Tool". In: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering book series (LNICST, volume 359), Springer-Nature, 2021.

Springer Link

International Conference on Computer Science and Health Engineering

↳ COMPSE 2020: **Computer Science and Health Engineering in Health Services** pp 231–241 | Cite as

Home > Computer Science and Health Engineering in Health Services > Conference paper

Teaching of TCP Fundamental Operations Using a Digital Tool

Ilse Alicia López-Pedroza & Francisco de Asís López-Fuentes

Conference paper | First Online: 25 February 2021

Part of the Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering book series (LNICST,volume 359)

Abstract

Today, digital tools play an important role in education. These tools are a support to strengthen the knowledge of students while for teachers it can become part of their work material. Using digital didactic tools, the students can understand an algorithm with a certain degree of difficulty. We present in this paper a digital tool to support the teaching and learning of transmission control protocol (TCP) fundamental operations. Using interactive examples and animations, our didactic tool provides complementary information that help to

Enlace: https://link.springer.com/chapter/10.1007/978-3-030-69839-3_16

5. López-Fuentes, F.A. Didactic Tool for Teaching Quality of Service Algorithms in Communication Networks, 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)


Conferences > 2022 IEEE 13th Annual Informa...

Didactic Tool for Teaching Quality of Service Algorithms in Communication Networks

Publisher: IEEE | Cite This | PDF

Francisco De Asis Lopez-Fuentes | All Authors

24 Full Text Views

Abstract

Document Sections

I. Introduction

II. Background

III. QoS Algorithms

IV. Operation of the Didactic Tool

V. Conclusions

Abstract:
Nowadays, information technology opens great opportunities to transform teaching and learning. Currently, different platforms have been deployed to offer teaching services. These platforms can be built from computer systems that collect and analyze data, to those systems that interact with users and allow them to learn from these data. For example, a system can build animations to visualize and understand the operation of an industrial process or an algorithm. This paper presents a digital tool to support the teaching and learning of quality of service (QoS) algorithms in the communication networks. Quality of service is an important topic in the computers networks courses due to the current high demand for quality multimedia content on the internet. Our digital teaching tool tries to support the teaching/learning processes in this important communication networks topic.

Published in: 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)

Enlace: https://ieeexplore.ieee.org/document/9946543

D. COMITE DE PROGRAMAS

1. Participé como Miembro del Comité Técnico de "The 12th Annual IEEE Information technology, Electronics and Mobile Communication Conference". 2021.

En esta conferencia se revisaron 5 artículos. (No se indican los nombres de los artículos por la confidencialidad del proceso de revisión). Se adjunta constancia de participación:

2. Miembro del Comité Técnico de “The 13th Annual IEEE Information technology, Electronics and Mobile Communication Conference”. En esta conferencia se revisaron 5 artículos. Se adjunta constancia de participación:

3. Revisor de 4 artículos científicos de la conferencia IEEE UEMCON 2021.

4. Se participó como revisor de un artículo (No se indica el nombre del artículo por la confidencialidad del proceso de revisión) para la revista Journal of Intelligent & Fuzzy Systems, en noviembre de 2021.

5. Revisor de 9 artículos científicos de investigación de la conferencia internacional 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC).

6. Integrante del jurado calificador del concurso para otorgar el Diploma a la Investigación 2020 de la División de Ciencias de la Comunicación y Diseño.

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

CAUC.170.21
9 de diciembre de 2021

A quien corresponda

En mi carácter de Secretario del Consejo Académico, hago constar que en la Sesión CUA-185-21 del Consejo Académico, celebrada el 31 de agosto de 2021 designó como integrante del Jurado Calificador del Concurso para otorgar el Diploma a la Investigación 2020 de la División de Ciencias de la Comunicación y Diseño al **Dr. Francisco de Asís López Fuentes**.

Cabe señalar que los integrantes del Jurado Calificador emitieron su dictamen el 5 de noviembre de 2021.

Se extiende la presente para los fines que al interesado convengan.

Atentamente
Casa abierta al tiempo

Dr. Gerardo Francisco Kloss Fernández del Castillo
Secretario del Consejo Académico

CONSEJO ACADÉMICO
Av. Vasco de Quiroga N° 4871, Torre III, 8° piso, Esq. Carlos Graef,
Col. Santa Fe Cuajimalpa, alcaldía Cuajimalpa de Morelos. C.P. 05348,
Ciudad de México. Tel. 58146506.
www.cua.uam.mx
otca@cua.uam.mx

ANEXOS:

Se anexan a este reporte:

A. Constancias de asesorías de Proyectos terminales concluidos
B. Constancias de Servicio social
C. Publicaciones

A. CONSTANCIAS DE ASESORÍAS DE PROYECTOS TERMINALES CONCLUIDOS

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

Comunidad académica comprometida con el desarrollo humano de la sociedad.

10 de septiembre de 2020

A QUIEN CORRESPONDA:

Por medio de la presente se hace constar que el profesor-investigador **Dr. Francisco de Asís López Fuentes**, asesoró a la alumna *Ilse Alicia López Pedroza (matrícula 209363342)*, en el desarrollo de su Proyecto Terminal titulado **Herramienta didáctica para la enseñanza del protocolo de comunicación TCP**, el cual concluyó satisfactoriamente en el trimestre 20-I.

Se extiende la presente para los fines que convengan al interesado.

ATENTAMENTE
Casa abierta al tiempo

Dr. Carlos Roberto Jaimez González
Coordinador de la Licenciatura en Tecnologías y Sistemas de Información
Correo: carlos.jaimez@gmail.com

UNIVERSIDAD AUTÓNOMA METROPOLITANA
UNIDAD CUAJIMALPA
Licenciatura en Tecnologías y Sistemas de Información

División Ciencias de la Comunicación y Diseño

Unidad Cuajimalpa
DCCD | Coordinación de la Licenciatura en Tecnologías y Sistemas de Información
Torre III, 5to. piso. Avenida Vasco de Quiroga 4871,
Colonia Santa Fe Cuajimalpa. Delegación Cuajimalpa de Morelos,
Tel. +52 (55) 5814-6550 y 51, C.P. 05348, México, D.F.
http://dccd.cua.uam.mx

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

División de Ciencias de la Comunicación y Diseño.
Licenciatura en Tecnologías y Sistemas de Información

Herramienta didáctica para la enseñanza del protocolo de comunicación TCP

Alumno: Ilse Alicia López Pedroza

Matricula: 209363342

Asesor: Dr. Francisco de Asís López Fuentes

Trimestre: 20-I

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

Ciudad de México a 28 de octubre de 2022.

Asunto: Constancia de Asesoría de Proyecto Terminal.

Por medio de la presente se hace constar que el Profesor Dr. **Francisco de Asís López Fuentes** asesoró el siguiente Proyecto Terminal de la Licenciatura de Tecnologías y Sistemas de Información que a continuación se detalla: **"Implementación de Codificación de red XOR usando FLUTER SDK para dispositivos móviles"**, el cual fue desarrollado por el alumno **Hossein Yahyasadeh, Matricula: 2183033723.**

Dicho proyecto culminó satisfactoriamente en el trimestre 22-P (23 de septiembre de 2022).

Se extiende la presente para los fines que convengan al interesado.

Atentamente
Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA
UNIDAD CUAJIMALPA
Licenciatura en Tecnologías y Sistemas de Información

Dr. Wulfrano Arturo Luna Ramírez
Coordinador de la Licenciatura de Tecnologías y Sistemas de Información
wluna@cua.uam.mx

Licenciatura en Tecnologías y Sistemas de Información

Unidad Cuajimalpa
DCCD | División de Ciencias de la Comunicación y Diseño
Coordinación de la Licenciatura en Tecnologías y Sistemas de Información
Torre III, 5to. piso. Av. Vasco de Quiroga 4871,
Colonia Santa Fe Cuajimalpa. Alcaldía Cuajimalpa de Morelos.
C.P 05348, Ciudad de México.
Tel.: (+52) 55.5814.6556
http://dccd.cua.uam.mx

Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

Unidad Cuajimalpa

División de Ciencias de la Comunicación y Diseño.

Licenciatura en Tecnologías y Sistemas de Información

IMPLEMENTACIÓN DE CODIFICACIÓN DE RED USANDO FLUTTER SDK PARA DISPOSITIVOS MÓVILES

Alumno: Hossein Yahyazadeh

Matricula: 2183033723

Asesor: Dr. Francisco de Asís López Fuentes

Trimestre: 22P

25/09/2022

B) CONSTANCIAS DE SERVICIO SOCIAL

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Cuajimalpa

CONSTANCIA DE ASESORIA DE SERVICIO SOCIAL
14 de Marzo, 2022.

A QUIEN CORRESPONDA:

La Sección de Servicio Social de la Universidad Autónoma Metropolitana Unidad Cuajimalpa, a través de la instancia de Servicio Social, hace constar que el profesor **DR. FRANCISCO DE ASIS LOPEZ FUENTES**, con No. Eco. 33210 adscrito al Departamento de **TECNOLOGIAS DE LA INFORMACION**, de la División **CIENCIAS DE LA COMUNICACION Y DISEÑO**, Unidad **CUAJIMALPA**, asesoró al siguiente alumno durante la prestación de su Servicio Social en el Proyecto: **Almacenamiento seguro de información en redes P2p usando técnicas de codificación de Red y disperción de información.**

	MATRICULA	NOMBRE DE ALUMNO LICENCIATURA DEPENDENCIA	FECHAS		
			INICIO	TERMINO	ACREDITACIÓN
1	2153076443	ORTEGA VALLEJO RAUL ANTONIO TECNOLOGIAS Y SISTEMAS DE INFORMACION UNIVERSIDAD AUTÓNOMA METROPOLITANA (UAM)	21/Jul/2020	27/Abr/2021	09/Jun/2021

Se expide la presente para los usos legales correspondientes.

Atentamente
Casa abierta al tiempo

LIC. MARIA DEL CARMEN SILVA ESPINOSA
JEFA DE LA SECCIÓN DE SERVICIO SOCIAL

UNIVERSIDAD AUTONOMA METROPOLITANA CUAJIMALPA SECCIÓN DE SERVICIO SOCIAL

Teaching of TCP Fundamental Operations Using a Digital Tool

Ilse Alicia López-Pedroza and Francisco de Asís López-Fuentes(✉)

Department of Information Technology, Universidad Autónoma Metropolitana - Cuajimalpa, Av. Vasco de Quiroga 4871, Cuajimalpa, 05348 México City, Mexico
ilsepedrozaa1@gmail.com, flopez@cua.uam.mx

Abstract. Today, digital tools play an important role in education. These tools are a support to strengthen the knowledge of students while for teachers it can become part of their work material. Using digital didactic tools, the students can understand an algorithm with a certain degree of difficulty. We present in this paper a digital tool to support the teaching and learning of transmission control protocol (TCP) fundamental operations. Using interactive examples and animations, our didactic tool provides complementary information that help to understand the TCP basic concepts.

Keywords: Didactic software · Networking · TCP protocol · Simulation

1 Introduction

During the year 2020 the world has faced a pandemic that kept us in social confinement, this includes the fact that schools have been closed for a long period. However, in this situation, alternatives were sought that could help distance learning. The main resource used to meet these needs have been digital tools and video conferences. The importance of technologies in education has been reflected, even increasing their price at the market and worldwide level. A greater benefit could be seen in education, some teachers realized that these technological resources could be used in face-to-face classes or for homework. The importance of didactic tools in teaching is reflected worldwide, it has already been accepted as a complement to the information given by teachers or as the means of teaching where teachers take the resources as the main source for their classes. A teaching tool helps support inside or outside the classroom, as it is easy to access and intuitive. However, the main limitation faced the digital tools based on internet is access to the internet since several localities still do not have the necessary internet infrastructure.

Networking courses are an important core of different bachelor curriculum related to information technology and computer engineering. However, motivating students to learn topics related to networking such as internet protocols can be often difficult and boring mainly due because its theoretical subjects. In this work, we present a digital tool to support the teaching/learning process of the fundamental operations of the Transmission

J. A. Marmolejo-Saucedo et al. (Eds.): COMPSE 2020, LNICST 359, pp. 231–241, 2021.
https://doi.org/10.1007/978-3-030-69839-3_16

Control Protocol (TCP) [1]. Using this tool, we expect that students can acquire, reinforce, and exercise their knowledge about this important protocol. Interactive examples and animations are used to guide the user through this tool.

The rest of this work has the following organization. Section 2 presents information related to protocols and networking didactic tools. In Sect. 3 information about basic and fundamental aspects of TCP are described. In this section, we also explain the TCP basic operations to be implemented in our didactic tool. Section 4 presents the general design of our didactic tool, while its implementation is described in Sect. 5. Section 6 describes the tests and evaluations done to our tool. The article concludes in Sect. 7.

2 Related Work

Networking protocols are described in many textbooks [2–6], and its applications have been discussed extensively in the computer networking literature [7, 8]. In this section, we review some didactic tools related to our work presented in this paper. Authors in [9] present NEO as a web tool where communication protocols can be learned. This tool mainly shows detailed definitions about communication protocols. There are animation sections where are shown the characteristics and operations of each OSI/ISO and TCP layer reference models. Isiunne is proposed in [10] as a methodology for the development of teaching/learning tools using animations. This tool has simple animations, which are used to explain data communication concepts, the ISO/OSI reference model and data communication protocols. Its design has been planned for different user profiles as administrator, teacher, or student. Isiunne contains evaluations of the concepts shown, which are administered depending on the user profile with which it is entered. Kiva Network simulator (KivaNS) is proposed in [11] as a free and open source java-based application. KivaNS allows user to design data network schemes and to simulate the IP routing through these networks. Thus, we can find data network diagrams and packet routing simulations. KivaNS is mainly oriented to simulate the IP behavior, and emulates the basic operation of technologies in the link layer as Ethernet. The main objective of this tool is to help to design and understand the data networks operations, specifically the packets routing in the TCP/IP architecture, without needing a real infrastructure and traffic analysis tools. Other digital tools related to teaching of computer networks can be found in the literature, however most of these tools still have many characteristics to be covered and improved.

3 TCP Background

The Transport Control Protocol (TCP) is a protocol in the transport layer, and its main task is the reliable transportation of data through the network [2]. This protocol allows to exchange information between computers and application programs. TCP ensures that the transferred data arrive correctly, secure and in order. Main attributes of TCP are [2]:

- a fully duplex bidirectional virtual circuit.
- data is transmitted as a data stream.
- its reliable data transmission is based on:

- sequence numbers
- checksum
- acknowledgements
- retransmission when an acknowledgement is received timeout.
- greater efficiency based the sliding-window principle.
- urgent data and push function
- graceful connection shutdown
- transport-user addressing using port number.

The TCP protocol header has different fields with the following significance [2], [5]: *source and destination port number*, are fields of 16-bits and denote the initial and end points of a virtual circuit. The *sequence number* is a 32-bits word which refers to the send direction. The *acknowledge number* also is a 32-bits word and it applies to the number of bytes received by the other end [2]. The *data offset* contains the length of the TCP header. The *flags* are bits used to trigger actions in TCP. There are six flag bits in the TCP header, and one or more of them can be activated at the same time [2]:

1. URG - pointer in *Urgent* field is valid.
2. ACK - acknowledgment number is valid.
3. PSH - the receiver should pass this data to the application as soon as possible.
4. RST - resetting of the connection.
5. SYN - synchronize sequence numbers to initiate a connection.
6. FIN – the sender shutdowns the connection, and the data flow is finished.

The *window* contains the number of bytes that a receiver can accept in its data buffer for this connection. This field is used to control the data flow. The *checksum* includes a code which is used in the receiver to detect some error in transmission. The *urgent pointer* points to a data byte called urgent data which must be read immediately. The *options* field is used to add extra facilities not covered by the regular header.

The TCP protocol contains many functions which ensure that the data arrives correctly. For example, TCP establish a successful and secure connection to ensure that information is not lost. The main functions to be considered in our didactic tool are TCP connection establishment (normal case), retransmission and sliding window. In the following we give a briefly description of these functions.

3.1 TCP Connection Establishment

A TCP connection is established through an agreement of three. One side waits for a connection while the other side executes the connection. For this connection, protocol needs to specify the address, the port number, the TCP maximum segment size to be accepted, and optionally some user data. After this, transmitter waits for the response. The segment arrives at the receiver and it checks if there is any process in the list. Depending on what is on the list, the connection is rejected or accepted. If the connection is accepted, then a confirmation segment is returned, otherwise a reject response is sent to the transmitter. A SYN segment consumes 1 byte of sequence space [5]. Figure 1 shows how a TCP connection is established.

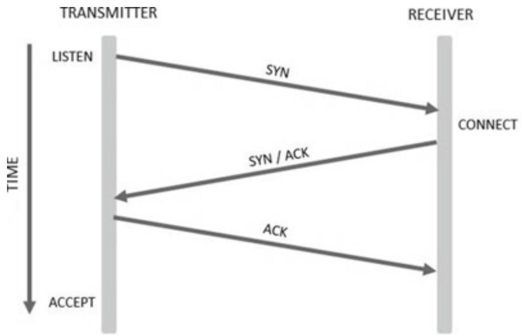


Fig. 1. Establishing a TCP connection.

3.2 Retransmission

During the information transmission may be loss of packets. Therefore, the retransmission is the way to ensure that the packets arrive correctly to its destination. TCP returns an ACK each time it receives data, on that path a timer is started. If the timer expires before the ACK arrives then the packet sending process is performed again. We can have *n* number of retransmissions, but all depends on the amount of lost packet that exist during the transmission. We can see this in Fig. 2.

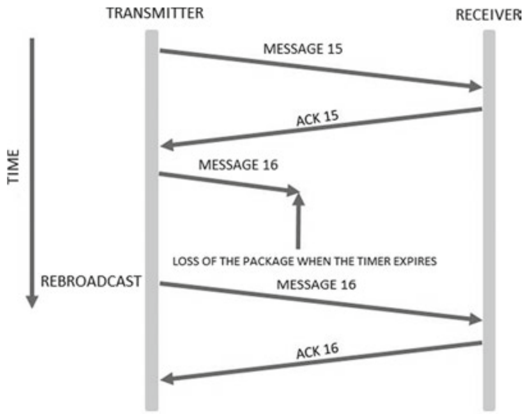


Fig. 2. An example of retransmission.

3.3 Sliding Window

TCP uses a sliding window scheme for data flow control in packet communications. Transmitter may send the number of bytes specified in the windows, without having to wait for an acknowledgement from the other part [2]. The sliding window can be enlarged or reduced depending on the needs. The aim is to carry the complete information from one side to the other. Figure 3 shows an example of how this function in the TCP protocol is done.

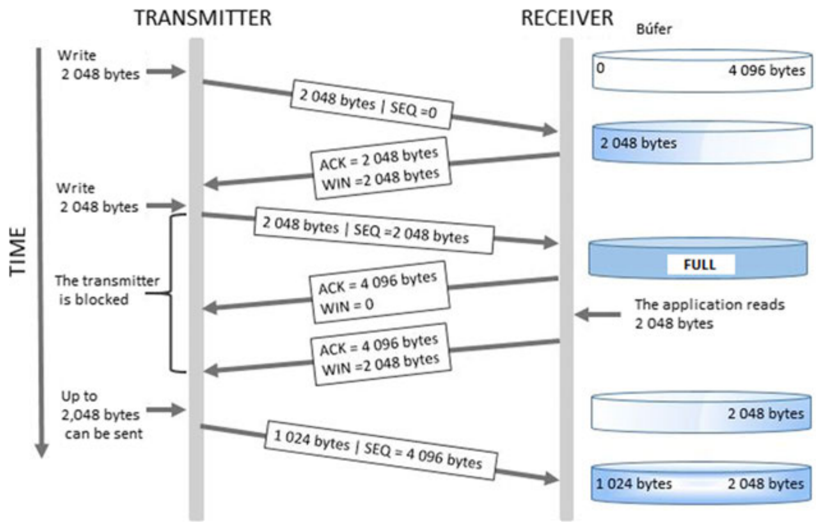


Fig. 3. An example of sliding window (adapted from [5]).

4 Analysis and Design

Designing an interface simple and easy-to-use both students and teachers is one of the objectives of this work. When a web page is created, we need to develop a system which should be easy to access and use by the main actors. In our case main actors are the students and teachers. This section describes the different design stages of our didactic tool.

4.1 System Structure

To develop our system, we have structured the TCP information into a main page, where each TCP function is represented. When the users enter the system, they can access all the information or only information that they could consider of interest. This is done through a main menu where the users can visualize this information.

4.2 Main Page

The main page shows the TCP definition and general information about this protocol. There is also a glossary with some technical terms that are used throughout the system. The main page of the system has a menu with three different options: TCP connection establishment, retransmission, and sliding window.

4.3 TCP Connection Establishment, Retransmission, and Sliding Window Sections

Any section in the main menu (connection establishment, retransmission or sliding window) has two options: definition and example. When the definition option is selected, we find a brief definition about the selected function and an animation. The animation shows a concrete example of how the function is performed between two devices and how the flow is carried out by this TCP function. The example option contains an interactive animation which represents a realistic case of how selected TCP function works (e.g.: connection establishment).

From these sections users can navigate through all the information in the system, they can learn about TCP connection establishment, retransmission or TCP sliding window. Even the users can return to the definitions or the examples.

In general, our design is proposed to use more images and animations avoiding use a lot of text. The animations invite to the user putting more attention in a specific TCP function. Thereby, our tool tries to capture the interest of students during the teaching/learning process. Likewise, teachers have a possibility to reinforce the knowledge given by them in the classroom. An approach based on animations can also be used to give repeated explanations about a topic or as a self-taught tool.

5 Implementation

Our didactic tool has been mainly developed using HTML and CSS language. Power Point and Scratch were used for the animations and examples. We use HTML to develop the system structure, the main menu was developed to display a submenu which shows the different options (definition and example) for each TCP function. We use CSS to define the colors and the fonts. Figure 4 shows the main page of the system where we can see at the top of menu the TCP functions (Our system has been written in Spanish).

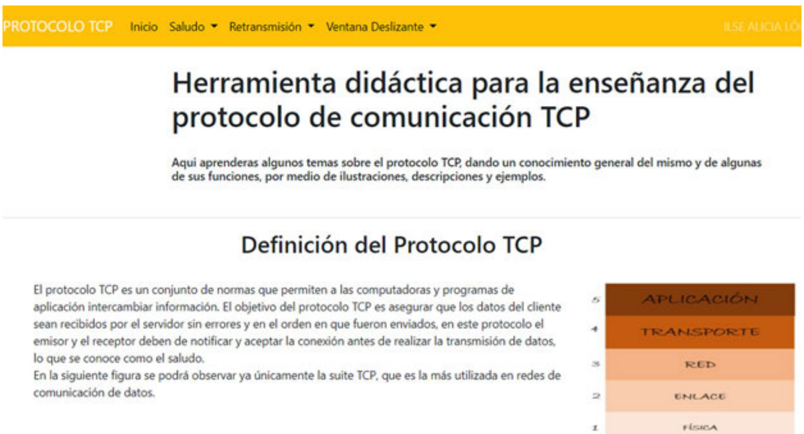


Fig. 4. System main page.

For a better understanding of the TCP, our system uses simple animations instead of images, since these animations explain step by step the processes of the TCP functions. First, the animations were made in Power Point, later they were integrated into the system using the iSpring tool. Using Power Point, we can draw the figures, add movements and more visual effects that help to understand the subject. We can obtain simple, understandable, and easy-to-use animations which can be viewed, stopped, restarted, or advanced. In this way, the user can see our tool as a more interactive method instead of an images-only collection. The Fig. 5 shows this animation development process and its final display in the system.

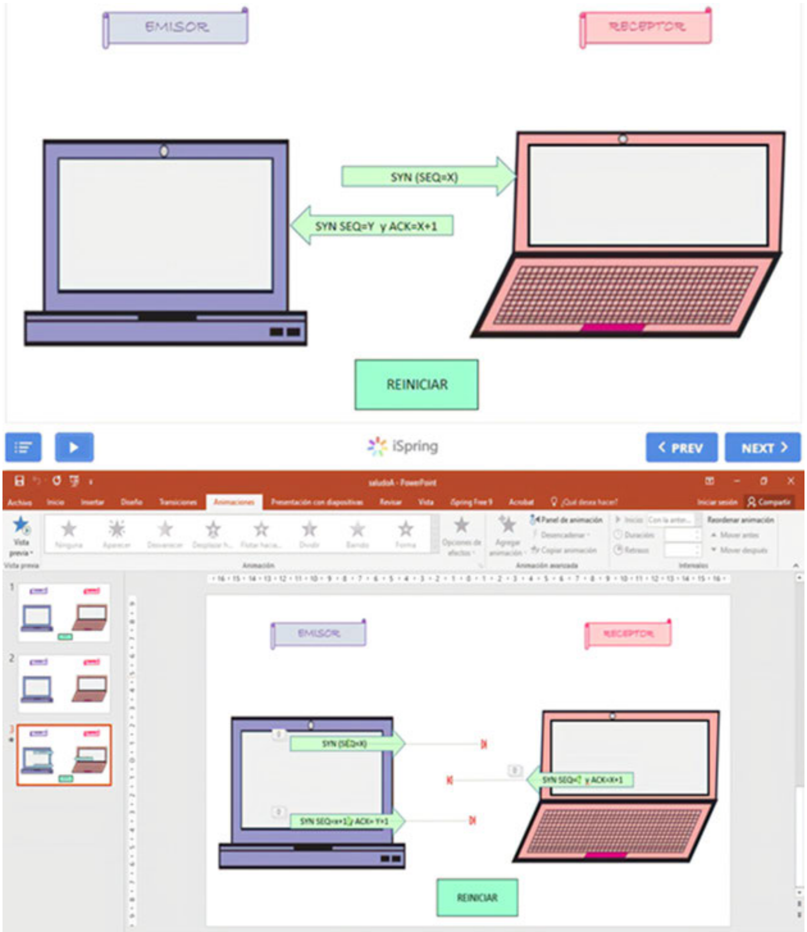


Fig. 5. Animation development for the connection establishment function.

The examples shown in the system are interactive, in the form of a game. So, the user can enter different data and simulate the moment when a failure in the function occurs or whether the function works correctly. Scratch [12] was used to develop the interactive examples. Scratch is a visual programming language, and it is an alternative to develop the examples in such a way that the general expectations of the system are met. The examples were programmed within the Scratch home page, so these examples can be accessed from this home page. Later the examples were added to the system. The examples can be reproduced as many times as the user wishes. Each time the user clicks on the green flag, the example will be started from the beginning, requesting the login data again. Figure 6 shows an example of how an interactive example for the retransmission function is displayed in the system.

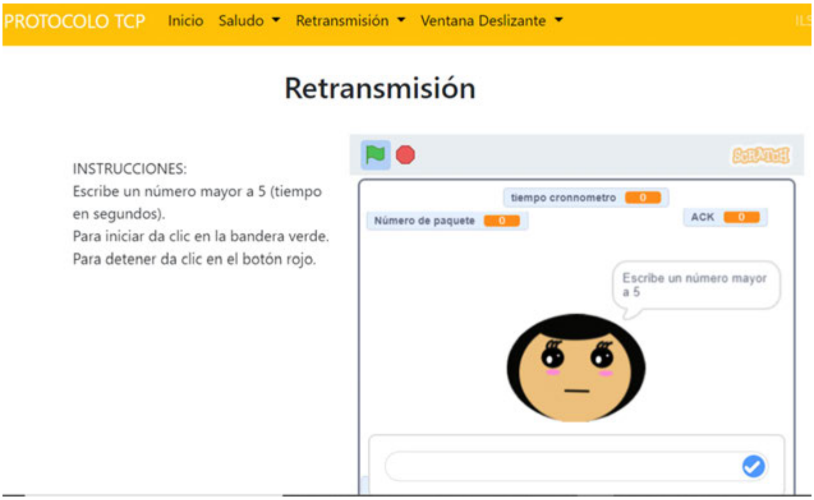


Fig. 6. Example of the retransmission function.

The programming of the animations was developed specifically according to the functions of the protocol. Figure 7 shows a general structure example about how the animations are displayed in Scratch. This application provides instructions, commands, variables, actions, functions, and other options. This instructions series are dragged to the middle, where the logic of programming the animations is carried out (actions in the right part of the Fig. 7). In this example we can see a programming block for the connection establishment function, which includes conditionals, and we can see that more than one block has already been added for a single drawing within the animation.

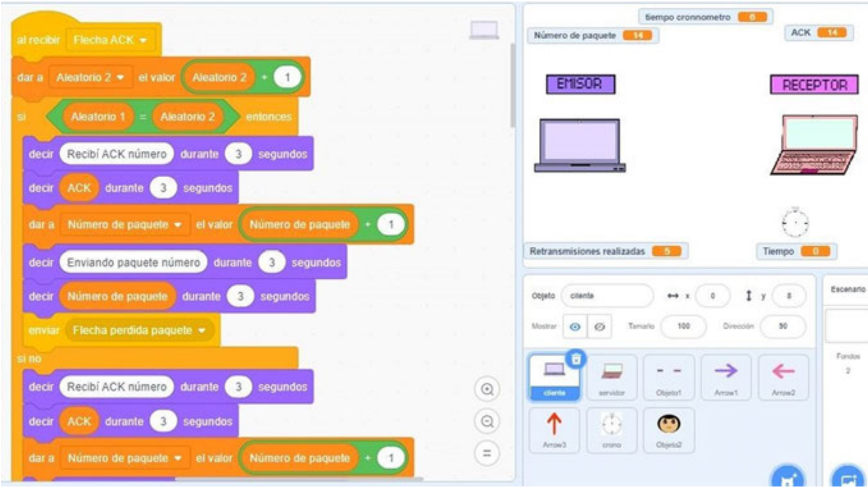


Fig. 7. Scratch development for the retransmission function.

6 Evaluation

Our didactic tool was tested with some users to evaluate its performance. The users have previous knowledge of TCP, some them are students and other graduates. Therefore, not all have recent knowledge over this protocol.

Users had access to the system, later they were given to fill out a form where the main questions were to highlight the design of the system and how this information was useful to them. The responses were very homogeneous since most of them had the same observations and likes for the system. The profile of the users who were tested was users with previous knowledge of TCP. The tests applied to users yielded results that helped to assess the quality of the system as well as see the advantages and improvements that can be made to it in the future work.

In general, the system was liked by the users. However, the design part received the most observations. The users made several observations related to the design and placing a greater emphasis on font size and system colors. Regarding the evaluation of the system, we observe that users showed a greater acceptance for the animations and interactive examples than the text. Some users did not use the text, because they were guided by the examples and animations. In fact, some users did not use the text, because they were guided by the examples and animations. Some results of our evaluation are shown in the Fig. 8.

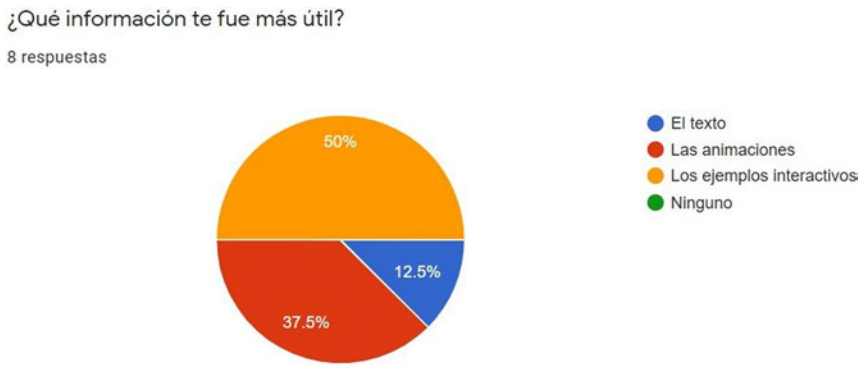


Fig. 8. User responses to the quality tests of our didactic tool.

7 Conclusions

The current world situation has shown us the importance of information technology, especially in education. In this paper, we present a didactic tool for teaching TCP protocol, which shows how information technology can play an important role in the teaching and learning process. The TCP protocol, like other subjects, can be taught through computational means such as cellular phones or electronic tablet. In our proposed didactic tool, we have learned that visual aspects are very important. The inclusion of little texts helps the users to have more taste and ease for learning of a subject. Our tool does not try to substitute to the teacher, but it tries to complement the information about TCP. Likewise, our tool could help to the students to clarify doubts related to TCP. Our didactic tool is easily accessible, it can be consulted within classrooms, at home or anywhere with internet access. A limitation of our tool is its dependence of internet because it cannot be used in location where the internet access is not available.

As future work, our didactic tool can be extended in different aspects. For example, we can add more animations, more extensive examples, new evaluation questionnaires, or aspect related to usability can be considered. Also, an open access version for teachers or students could improve this tool with feedback or collaboration. Finally, others TCP functions can be added such as round-trip time during the retransmission or a comparison with UDP communication protocol.

References

1. Postel, J.: Transmission Control Protocol. RFC 793 (1981)
2. Santifaller, M.: TCP/IP and NFS Internetworking in UNIX environment, Addison-Wesley (1981)
3. Peterson, L.L., Davie, B.S.: Computer Networks: A Systems Approach. 5th edition, Morgan Kaufmann (2011)
4. Kurose, J., Ross, K.: Computer Networking: A Top-Down Approach. 7th edition, Pearson (2016)
5. Tanenbaum, A., Wetherall, D.J.: Computer Networks. 5th edition, Pearson (2011)
6. Comer, D.E.: Computer Networks and Internet. 1st edition, Prentice-Hall (1997)

7. Stalling, W.: Data and computer communications. 8th edition, Pearson Prentice Hall (2007)
8. Comer, D.E., Stevens D.L.: Internetworking TCP/IP Vol. 2, 3th edition, Prentice-Hall (1995)
9. NEO: http://neo.lcc.uma.es/evirtual/cdd/tutorial/Indice.html. Accessed 08 Dec 2019
10. Isiunne: http://www.lsiunne.com.ar/_. Accessed 10 Dec 2019
11. KivaNS: http://www.disclab.ua.es/kiva/. Accessed 10 Dec 2019
12. Scratch: https://scratch.mit.edu/. Accessed 22 Jul 2020

Network Coding and Dispersal Information with TCP for Content Delivery

Francisco de Asís López-Fuentes[1](✉), Raúl Antonio Ortega-Vallejo[1], and Ricardo Marcelín-Jiménez[2]

[1] Universidad Autónoma Metropolitana-Cuajimalpa, Av. Vasco de Quiroga 4871, Cuajimalpa, 05348 Mexico City, Mexico
flopez@cua.uam.mx, raulantonio@protonmail.com
[2] Universidad Autónoma Metropolitana-Iztapalapa, Av. Atlixco 186, Iztapalapa, 09340 Mexico City, Mexico
cal@xanum.uam.mx

Abstract. Dissemination information from many sources to many receivers can be fundamental in different systems. However, the components of these system may present some failure type both the software and hardware. In addition, problems related to the communication networks such as limited bandwidth or packet loss should be present. The information dispersal algorithm (IDA) has been used as a good solution to offer fault tolerance. On the other hand, network coding is a coding method mainly used to increase throughput of a communication channel, which is useful to face the limited bandwidth in the communication networks. In this paper, we integrate both methods into a content distribution scheme. We use a hybrid peer-to-peer (P2P) network based on TCP in order to evaluate the performance of IDA and network coding in a joint operation.

Keywords: Network coding · Information dispersal · P2P networks · TCP

1 Introduction

Content delivery is very popular today, users can exchange several content types such as video, text messages, PDF documents, music, and photos. Some content such as video demand for a large amount of resources from the Internet infrastructures and cooperation between nodes play an important role. New problems such as fault tolerance, limited performance or bandwidth limitations have emerged. To face these challenges some techniques such as the information dispersal algorithm (IDA) [1] and network coding [2] have been proposed. Using IDA in the communication systems we can reach redundancy of the information in different levels and make better use of the storage capacities of the devices. In this way, we can configure fault-tolerance storage systems more efficient. On the other hand, network coding allows that the intermediate nodes encode the received packets for immediately forwarding the encoded packets to the end nodes [3, 4] and [5]. In other words, the packets received in the intermediate nodes are combined before

J. A. Marmolejo-Saucedo et al. (Eds.): COMPSE 2021, LNICST 393, pp. 63–72, 2021.
https://doi.org/10.1007/978-3-030-87495-7_5

forwarding them to the following nodes. Network coding uses elements of a finite field for the linear operations during the packet manipulation. The communications systems can be benefited by using network coding because this technique helps to increase the throughput and reduce the latency. Although different benefits by using network coding and IDA are reported in the literature [18, 19], both techniques have been used separately. This paper proposes an architecture where the IDA and network coding are combined. Peers are used as a way to improve the content delivery. Our proposed architecture is implemented using TCP. We use specific characteristics of this protocol such as the retransmission as a way to reduce the number of descriptors to be sent to each intermediate node.

This paper has the following organization. Section 2 presents related work to IDA and network coding. Section 3 introduces basic and fundamental aspects of network coding and IDA. In this section, we also explain some concepts about P2P networks and TCP. Section 4 presents our proposed architecture, where network coding and the information dispersal algorithm are combined under TCP. Our implementation and an initial evaluation is described in Sect. 5. Our conclusions are presented in Sect. 6.

2 Related Work

Several studies to address the impact of redundancy on P2P storage systems have been reported in the scientific community. Many of these works have related with network coding and the information dispersal algorithm (IDA). The works of [10, 11, and 12] evaluate redundancy assuming a static network. In [10], there is a thorough analytical study of the mean-time to failure (MTTF), bandwidth and storage load, for either simple replication or error-codes, particularly based on Rabin's IDA [1]. Failures on the set of storage devices are regarded as independent and identically distributed (iid) random variables. The main result shows that, compared to simple replication, IDA provides by far a higher MTTF, under the same amount of redundancy. Also, IDA requires less bandwidth and storage. In turn, [11] is focused on the excess of information required to different strategies, in order to provide the same level of availability for a collection of files. This study is based on Monte-Carlo simulations. Each experiment consists of a number of files which are processed according to a given strategy, and require allocation on the devices that make up the system. Nevertheless, devices are available with a given probability. The main result here is that IDA needs less space to provide the same level of availability, compared to simple replication. The second part of [11], is an effort to consider the long term behavior of a P2P storage network. For this purpose, any peer connection is assumed to last an exponential random time. The findings show that the files' availability presents a faster degradation when stored under IDA replication. In [12], the study compares three strategies: *uncoded random storage, traditional erasure coding based storage, and random linear coding based storage.* The result proposes that network codes are a good option to maintain the best efficiency of the system. Two important studies [13, 14] developed models that evaluate the amount of redundant information delivered by either simple replication or IDA, according to the level of availability required by the set of allocated files. Also, both works addressed the problem of maintainability, i.e. the cost of recovering and reallocating the information stored in a peer which is presumed to be left permanently. They introduced the concept of membership expiration time, in order to estimate whether a disconnection should be regarded as temporal or permanent. Using this parameter they developed a formula to evaluate the amount of average peer bandwidth required to keep any file within a given level of availability. Apart from simple replication, IDA and the hybrid approach by [13], the work of [15] introduces network codes. The study is based on analytical models and trace-driven simulations. Their findings show that network codes provide a very efficient mechanism to support information maintenance. Authors in [18] use IDA to make an efficient content delivery on a P2P network. A solution for a secure PLC (Power Line Communication) communication among end nodes based on the Information Dispersal Algorithm (IDA) is presented in [20]. In this work, the authors propose an efficient scheme using the physical characteristics of PLC channels of a smart grid.

On the other hand, network coding have used in several works for content delivery, because this technique has already proven to provide solutions to a variety of networking problems. For example, authors in [16] presents a unified linear program formulation for optimal content delivery in content delivery networks (CDNs). In this work, different costs and constraints associated with content disseminations are considered from the source node to storage nodes. The end users can do an eventual fetching of content of storage nodes. In [17] network coding is deployed in the backbone network for an IPTV architecture. In this case, network coding helps to increase network capacity while improving robustness against network faults. Authors in [21] introduce ND-POR, which is a scheme based on network coding but it is observed that these can be sensitive to a certain type of small corruption attack on their integrity and, to turn it around, the dispersal coding is applied. This proposal has the main target the cloud storage systems. To the best of our knowledge, in most of the literature reviewed, both network coding and IDA have been used separately. In this paper is presented a collaborative architecture which combines network coding and IDA with TCP over a hybrid P2P network in order to evaluate the performance of both technique during content delivery. Peers help to improve the packets delivery, while TCP allows to reduce the number of descriptors to be sent to the intermediate nodes.

3 Background

Information redundancy is an important concern of any communication system. This important characteristic can be reached via replication or coding. However, both techniques have different approach. For example, using replication we can allocate 3 copies of a file F, on 3 different storage sites. In this way, a failure in 2 storage sites can be tolerated by the system. Nevertheless, the effective storage capacity is only one third of the overall capacity. Now, we review information redundancy using IDA [1]. In this case, file F is transformed into n dispersals, each one of size $|F|/m$ and F can be recovered provided that any m out of n dispersals remain available (see Fig. 1). We can see in Fig. 2 that it is only required n/m $\times$ 100% of the original file size as redundant information using IDA. Let us suppose that we use a particular implementation of IDA with parameters n = 5, and m = 3. Compared to the previous case of replication, the file can be transformed into n = 5 dispersal, each of size |F|/3. In this case is produced an excess of information of (5/3) × 100%, which is less than 300% of information required by the replication strategy. Thus, any 2 dispersals can be loosed, but it is still possible to recover the original file. Based on this approach, IDA presents an effective capacity almost twice compared with the replication strategy. This scenario is shown in Fig. 2.

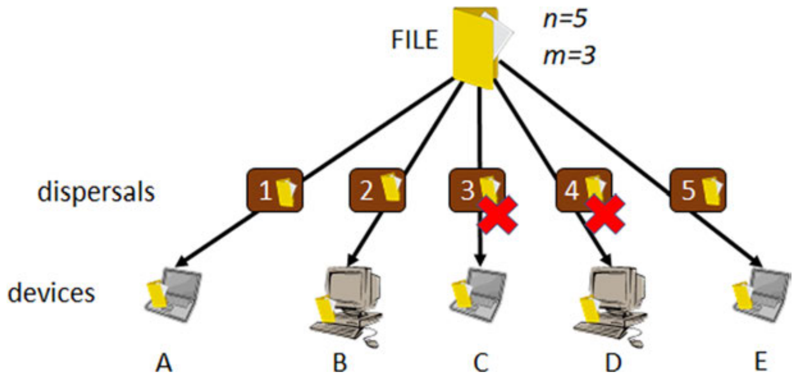


Fig. 1. IDA concept.

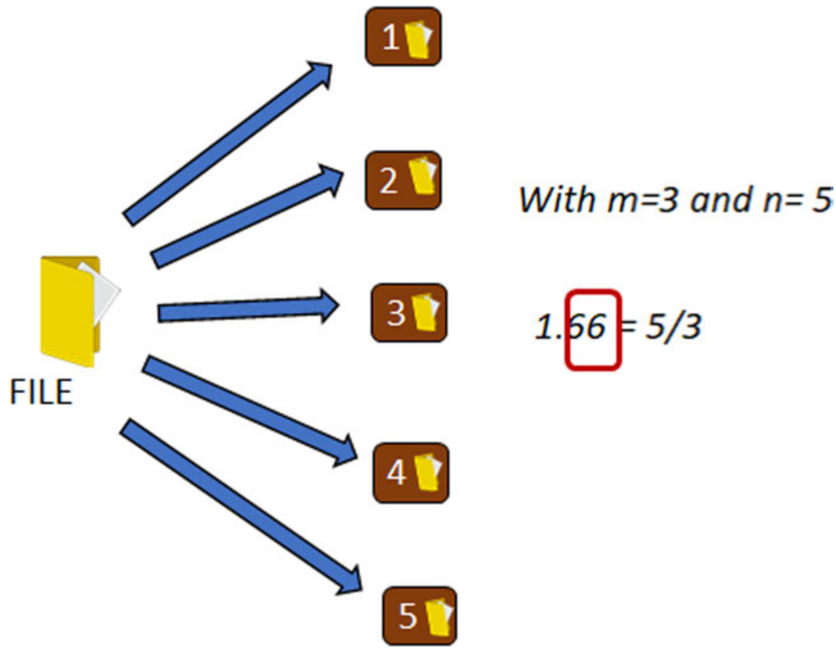


Fig. 2. An example of IDA for m = 3 and n = 5.

Network coding is an encoding technique used to increase the flow of packets without exceeding the link capacity [2]. To explain this scenario, we use a butterfly network (see

Fig. 3), which has a source node S, and two receiving nodes R1 and R2. Each edge has capacity of 1 as shown in Fig. 3a, and we can see that the maximum flow from the source S to any receiver (R1 or R2) has a value of 2. Simultaneously, source S sends bit b1 to receiver R1 and bit b2 to R2 (see Fig. 3b). Node 3 receives bits b1 and b2 from nodes 1 and 2, respectively, and it must send both bits to node 4. However, the link between nodes 3 and 4 requires two time units to send both bits. In contrast, using network coding (which is indicated with the operator $\oplus$ in Fig. 3c) the receiver R1 can recover both bits (b1 and b2), but bit b2 must be recovered from operation b1 $\oplus$ b2. Receiver R2 recovers both bits making a similar procedure as R1. In this scenario, node 3 is responsible to apply network coding. We can note that network coding allows to increase the multicast rate in the link from 1 to 2 bits/time unit.

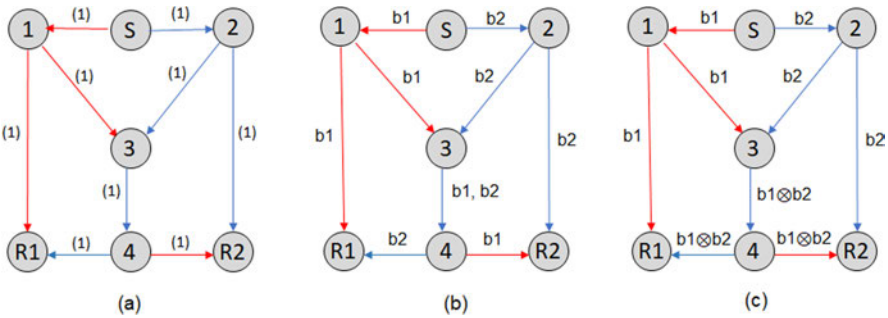


Fig. 3. Example of a communication network. a) Capacity of the edges, b) Traditional approach and c) approach with network coding.

P2P networks have become a popular paradigm for the next generation of distributed computing, and they are used to spread digital content to a large audience [6]. These types of networks are a research topic in several areas such as data communication networks, distributed systems, complexity theory and databases [7]. A P2P network is a virtual communication infrastructure deployed over a physical network. Nodes build a network abstraction on top of the physical network, and it is known as an overlay network. This overlay network is independent of the underlying physical network, and the connections between nodes are done using the Transmission Control Protocol (TCP). This protocol allows to abstract the physical connections in such a way that these are not reflected in the overlay network. The routing mechanisms use the logical tunnels implemented between nodes by the overlay network [8]. TCP is a transport protocol used to provide reliable delivery of data via a communication network [9]. Computers can exchange data with application programs in a way correct, secure and in order by using TCP. Communication networks can present packet loss during a transmission. To deal with this problem TCP uses retransmission to ensure the delivery of the packets. TCP can do n number of retransmissions, depending on the number of lost packets during the transmission. In this work, we combine IDA and network coding using TCP over a P2P infrastructure.

4 Proposed Model

This section describes our proposed model. Figure 4 shows our architecture, which is formed by 15 different nodes (peers and servers).

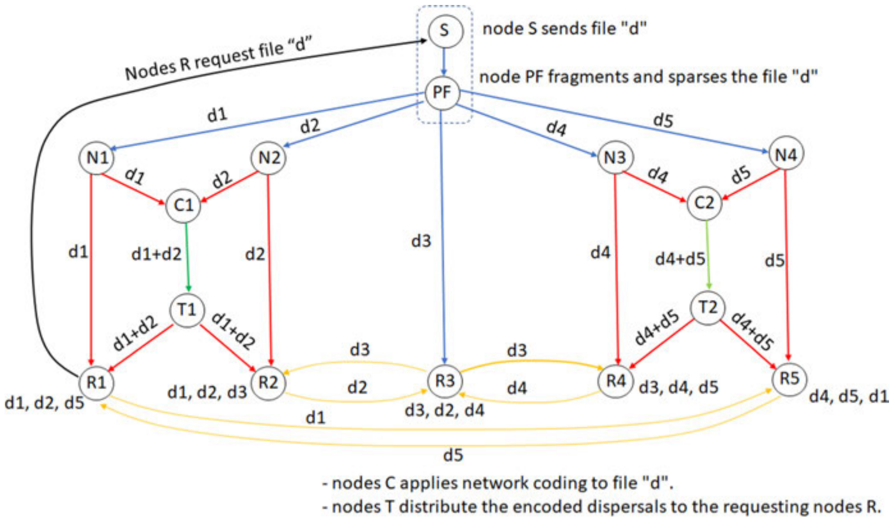


Fig. 4. Architecture combining network coding and IDA.

Node S is the source node, while node PF is the fragmenting peer, which is the responsible to fragment the file using the information dispersal algorithm (IDA). We use two butterfly schemes to deploy network coding. The nodes N1 and N2 are the source nodes in the butterfly 1, while N3 and N4 are the source nodes in the butterfly 2. Nodes C1 and C2 are the intermediate nodes which are responsible to do the network coding operation in the butterfly 1 and 2, respectively. Node T1 relays the encoded message received from the node C1 to the requesting nodes R1 and R2 in the butterfly 1, while node T2 relays the encoded message received from the node C2 to the requesting nodes R4 and R5 in the butterfly 2. The requesting peer R3 receives the descriptor from the node PF directly. All requesting nodes R are working as peers, which establish communication between them to share the descriptors and to have information about the contents that are shared within the network. For each delivery, a peer R creates a thread to distribute the received descriptor to other peers. The communication between a sending peer and a requesting peer is established through this thread. All nodes in the architecture have a specific role during the transmission, therefore they are renamed before that a file be sent to a requesting peer. All peers (nodes R) collaborate with each other to distribute a descriptor in our architecture. That is to say, the peers work as relay node too. This means that a peer receives a dispersal from the fragmenting node (PF) or node T and retransmits this dispersal to other peers in the architecture. For example, node R3 receives descriptor d3 from node PF, then d3 is relayed to nodes R2 and R4. On the other hand, node R1 receives descriptor d1 from node N1, then d1 is relayed to node R5. Nodes R2, R4 and

R5 have a similar behavior as node R1 because they only distribute a descriptor. Our proposed architecture works as a multi-source scheme because the nodes R receive three different dispersals from three different nodes.

Initially, the source sends the requested file by the requesting nodes R to the fragmenting node, which uses the IDA algorithm to fragment the file into five descriptors. Each receiver node R can recover the original file having only three descriptors. The fragmenting node requests the IP addresses of the nodes (N1, N2, R3, N3 and N4). Each of these IP addresses receives a descriptor from The PF node. After this, nodes N work as sources, and network coding is applied using two butterfly schemes. Nodes R1, R2, R4 and R5 recover two descriptors by decoding the message received from the nodes T. These receiving nodes with two descriptors establish communication with the other requesting nodes to delivery their descriptors. Because node R3 only receives a descriptor, this node should receive a descriptor from nodes R2 and R4. Thus, all requesting nodes R obtain the missing descriptor and they can assemble the original file using the IDA algorithm.

5 Implementation and Evaluation

Our work is in progress, and we have done an initial implementation of our architecture. This prototype has been developed in the C programming language for the Debian Linux operating system. Our prototype uses 14 containers and we performed 6 runs for each experiment. We transmitted different source vectors with the following dimensions: 1 MB, 5 MB, 10 MB, 15 MB, and 30 MB.

Our first experiment evaluates a content delivery using IDA without network coding. In this case, the butterflies are not done, and the nodes R receive the descriptors directly from the nodes N. Thus, the most intensive interaction occurs between the R nodes and the nodes N, which must retransmit the flow from the node PF. Nodes N work as relay nodes. Each vector is simulated as a broadcast, where because TCP is used. The nodes N require concurrent processes to emit a vector "d". The situation is similar for node PF when issuing all the vectors resulting from the IDA algorithm. In the second experiment, the topology shown in Fig. 4 is evaluated. In this case IDA and network coding are combined through the implementation of two butterfly schemes. Each butterfly applies network coding for the data streams coming from node PF. The nodes R work as peers and transmit the dispersal in a collaborative way. The network coding scheme used was the traditional network coding scheme, which is based on the use of arithmetic operations such as the binary XOR operation.

We measured the execution times for both experiments using two methods. The first method counts the time required by the processor, while the second method counts the real time of the simulator application, giving an expectation of the possible waiting time scenario for a user. In this way, the receiving nodes, which are the first nodes to be executed, accumulate the logs of the reception times. Results are shown in Figs. 5 and 6. We can see in Fig. 5 (IDA over TCP) the time costs for the execution of the simulator in simple transmissions of the scattered vectors. For a file of 1MB, we obtain a time of 0.632844 s for the processor, and a total time of 12.77619012 s (real time), which is the time it took for the simulator program to recover the complete source vector between each receiver. Figure 6 shows the results obtained from experiments using IDA with network coding. Here, we can observe that the transmission of a 1MB vector requires a processor time of 0.616914 s, while the total real time for the execution and termination of the simulator is 6.31214858 s.

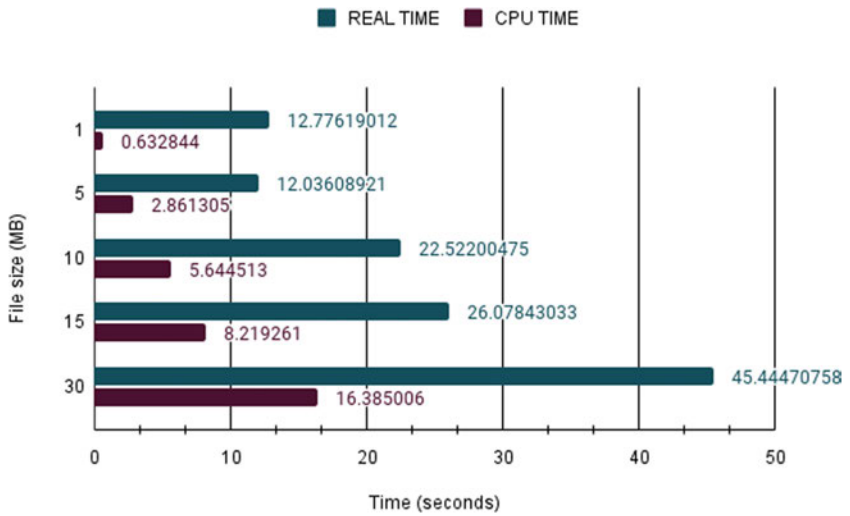


Fig. 5. Comparison of distribution time for architecture with IDA only.

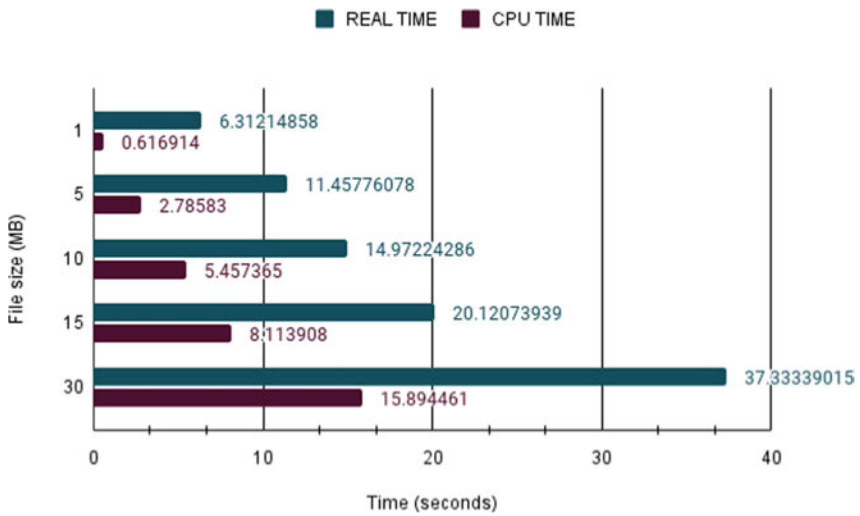


Fig. 6. Comparison of distribution time for architecture with IDA and network coding.

The used methods show that the CPU times could increase with the use of the network coding, because the intermediate nodes require more processing to perform the necessary operations of the schema. This could mean that the proposed architecture may generate a higher cost over time. However, the figures of real seconds, allow to visualize an expectation of the real time for the users (nodes R in Fig. 4) of the architecture. On the other hand, delivery time variations can be observed for different sizes of the source vector. We can also observe that for small vectors (e.g.: 1 MB and 10 MB) it might be convenient to use the proposed architecture. We are evaluating the performance of this architecture using larger vectors.

6 Conclusions

Content delivery plays an important role in the current communication networks. Because these services demand efficient distribution schemes. Network coding and the information dispersal algorithm (IDA) are techniques used to mainly improve the throughput of the communication networks and fault tolerance in the storage systems, respectively. Several applications using both techniques can be found in literature, but separately. In this work, we propose a distributed architecture which combines network coding with IDA to evaluate the impact of this strategy. Our work is in progress, and the preliminary results show that network coding combined with IDA can reduce the delivery time for files of small size. Our implementation has been done using TCP. Therefore, some specific characteristics of TCP such as retransmission of loss packets is very important. Requesting nodes working as peers play an important role because it allows these to work as relay nodes and not as simple leaves of a tree.

As future work we plan continue our experiments evaluating larger files to observe the behavior of our architecture under these scenarios. In addition, we are working in an implementation of our architecture using UDP in order to compare its performance with our implementation done in TCP. Data privacy can also be implemented in our architecture by using encryption techniques based on the AES algorithm.

References

1. Rabin, M.O.: Efficient dispersal of information for security, load balancing, and fault tolerance. J. ACM **36**, 335–3348 (1989)
2. Ahlswede, R., Cai, N., Li, S.-Y., Yeung, R.W.: Network information flow. IEEE Trans. Inf. Theory **46**, 1204–1216 (2000)
3. Chou, P., Wu, Y., Jain, K.: Practical network coding. In: 51st Allerton Conference on Communication, Control and Computation, Monticello, IL, USA (2003)
4. Gkantsidis, C., Rodriguez, P.R.: Network coding for large scale content distribution. In: IEEE INFOCOM 2005, Miami, FL, USA, pp. 2235–2245 (2005)
5. Sundararajan, J.K., Shah, D., Medard, M., Mitzenmache, M., Barros, J.: Network coding meets TCP. In: IEEE INFOCOMM 2009, Rio de Janeiro, Brazil, pp. 280–288 (2009)
6. Androutsellis-Theotokis, S., Spinellis, D.: A survey of peer-to-peer content distribution technologies. ACM Comput. Surv. **36**(4), 335–371 (2004)
7. Milojicic, D.S., et al.: Peer-to-peer computing. Technical report HPL-2002-57R1, HP Labs, Palo Alto (2002)
8. Ripeanu, M., Foster, I., Iamnitchi, A., Rogers, A.: In search for simplicity: a self organizing multi-source multicast overlay. In: 1st IEEE International Conference (SASO 2007), Boston, MA, USA, pp. 371–374 (2007)

9. Santifaller, M.: TCP/IP and NFS Internetworking in UNIX Environment. Addison-Wesley, Boston (1981)
10. Weatherspoon, H., Kubiatowicz, J.D.: Erasure coding vs. replication: a quantitative comparison. In: Druschel, P., Kaashoek, F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 328–337. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45748-8_31
11. Bhagwan, R., Moore, D., Savage, S., Voelker, G.M.: Replication strategies for highly available peer-to-peer storage. In: Schiper, A., Shvartsman, A.A., Weatherspoon, H., Zhao, B.Y. (eds.) Future Directions in Distributed Computing. LNCS, vol. 2584, pp. 153–158. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-37795-6_28
12. Acedanski, S., Deb, S., Médard, M., Koetter, R.: How good is random linear coding based distributed networked storage? In: 1st Workshop on Network Coding, Theory, and Applications (NetCod) (2005)
13. Rodrigues, R., Liskov, B.: High availability in DHTs: erasure coding vs. replication. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 226–239. Springer, Heidelberg (2005). https://doi.org/10.1007/11558989_21
14. Blake, C., Rodrigues, R.: High availability, scalable storage, dynamic peer networks: pick two. In: HOTOS 2003: Proceedings of the 9th conference on Hot Topics in Operating Systems, pp. 1–6 (2003)
15. Dimakis, A.: Network coding for distributed storage systems. In: INFOCOM 2007. 26th IEEE International Conference on Computer Communications, pp. 2000–2008 (2007)
16. Derek, L., Ho, T., Cathey, R.: Optimal content delivery with network coding. In: 43rd Annual Conference on Information Sciences and Systems, CISS 2009, pp. 414–419 (2009)
17. Kwon, M., Kwon, J., Park, B., Park, H.: An architecture of IPTV networks based on network coding. In: International Conference on Ubiquitous Future Networks ICUFN, pp. 462–464 (2017)
18. López Fuentes, F.A., Mendoza Almanza, J., Marcelin-Jiménez, R., Velázquez-Méndez, B.: Efficient content distribution and storage P2P system based on information dispersal. In: 6th International Conference on Control, Decision and Information Technologies (CoDIT) (2019)
19. Mendoza-Almanza, J., López-Fuentes, F.A.: Optimal network coding based on machine learning methods for collaborative networks. In: 6th International Conference on Control, Decision and Information Technologies (CoDIT) (2019)
20. Noura, H.N., Melki, R., Chehab, A., Hernandez-Fernandez, J.: Efficient and robust data availability solution for hybrid PLC/RF systems. In: Computer Networks (2021)
21. Omote, K., Tran, P.T.: ND-POR: a POR based on network coding and dispersal coding. IEICE Trans. Syst. **E98-D**(8), 1465–1476 (2015)

Content Distribution and Storage Based on Volunteer and Community Computing

Francisco de Asís López-Fuentes(✉)

Department of Information Technology, Universidad Autónoma Metropolitana - Cuajimalpa, Av. Vasco de Quiroga 4871, Cuajimalpa, 05348 México City, Mexico
flopez@cua.uam.mx

Abstract. In recent years, the users need to discover and use a diversity of resources in the Internet to do their tasks. These resources such as massive storage, processing and distribution capacity are generally decentralized and geographically dispersed, however they can be shared to solve large-scale problems in a collaborative way. Peer-to-peer (P2P) networks are an attractive alternative to implement collaborative solutions. This work presents a P2P collaborative for content distribution and store management in small communities based on volunteer and community computing.

Keywords: P2P networks · Collaboration · Distributed systems · Simulation

1 Introduction

Information and communication technologies have significantly altered the ways people communicate, entertain, work, negotiate, govern, or socialize on a global scale. The Internet, better computing and communication capabilities have led to an interest in decentralizing and sharing geographically dispersed resources to solve large-scale problems. Under this approach, various computing resources such as processing capacity, massive storage and high capacity networks are offered as services. Resource management and application programming in large-scale distributed systems is a hard task, since different scenarios such as the variable number of available resources and the different requirements defined by users must be evaluated to improve their performance. However, users do not know how the computational structure of these systems have been constructed because a resource agent hides this complexity to the users during the interaction. Many of these services are based on centralized approaches, which introduce various limitations related to locality, dependency and a single point of failure.

Collaborative computing infrastructures such as peer-to-peer networks have emerged as an important solution for managing distributed resources on the Internet, but they still have open problems [16]. The Internet has triggered a social revolution due to the way in which people interact with each other on a planetary scale. Thus, people with common interests but geographically separated can create small communities around the world for cooperation between them. This approach can be based on community and volunteer computing. Community computing is a model where all computing services

M. F. Mata-Rivera and R. Zagal-Flores (Eds.): WITCOM 2021, CCIS 1430, pp. 163–173, 2021.
https://doi.org/10.1007/978-3-030-89586-0_13

are based on a cooperative resource-sharing approach between users [1]. Community computing is studied by many researchers for several applications mainly related with cloud computing [3, 4]. For example, community cloud can be used by organizations with the same requirements and needs in order to save costs [25]. On the other hand, voluntary computing refers to idle computing resources that are shared by public participants to support computationally expensive projects [2]. Volunteer computing uses the distributed resources as an important strategy to do large-scale tasks [5] such as potential processing and storage [6, 7]. For example, the users of a social network may share their heterogeneous computing resources to form a in a social ad hoc could [24].

Although community and voluntary computing have been studied during the last years, these paradigms still presents different challenges and opportunities. Future directions related to cloud and fog computing, security, privacy, and reliability are open. Also, new applications based on community and voluntary computing for organizations with common interests may be deployed. In the present work, we propose a P2P collaborative infrastructure as a possible solution for supporting the generation and dissemination of contents in small communities. Our main motivation in this work is to create an open collaborative platform where we can evaluate new protocols and algorithms for the distribution and storage of content. Our second motivation is to propose an open architecture to small communities with common interests that want to share their resources without using commercial applications.

Our paper continues with the following organization. Information about community and volunteer computing architectures is presented in Sect. 2. Concepts related to P2P networks are described in Sect. 3. In Sect. 4, we present our proposed P2P collaborative architecture. Because our work is in progress, in Sect. 5 we present a basic evaluation of our proposed architecture to know its impact in terms of collaboration. The article concludes in Sect. 6.

2 Related Work

Peer-to-peer networks are a promising paradigm in the distributed computing [10], and it has been used as a way to spread digital content to a large audience [9]. Interest in this paradigm is growing because they allow geographically distributed computational resources to be coupled to solve large-scale problems. These new approaches help to solve complex problem in several areas related to science and engineering [8] because the data and computational resources can be exchanged, selected aggregated regardless their physical location. P2P systems still present open issues and research opportunities, and P2P networks has been used to deploy IoT (Internet of Things) [17], blockchain [18] or grid solutions [19].

Authors in [1] propose a cooperation scheme using community computing in order to model and describe cooperation effectively. This model gives an introduction about the conflict resolution in the community computing. Authors presents some case studies to evaluate the community computing and its model. Babaoglu et al. [20] introduce a P2P architecture for providing cloud services. This work presents how a fully decentralized P2P cloud has been designed and implemented. A distributed computing infrastructure allows for organizations and individuals to use existing resources for allocating different tasks. Main problem to been solved is the coherency of the structure under unreliable conditions of the computing resources. Overlay network is constructed on top of the physical network using gossip-based communication protocols. Resources are partitioned into multiple slices in order to avoid that individual failures in a node affect the overall network. The resource partitioning process receives special attention in this work, because it must be conducted in an efficient and reliable way in any cloud architecture. A prototype of this architecture was implemented as a way to demonstrate the effectiveness of this proposal.

Marzal-Romeu et al. in [19] review information and communication technologies related to the microgrids. In this work is investigated which is the most suitable network topologies and protocols for smart microgrids. Because microgrids and P2P networks have a dynamic behavior, the authors conclude in their study that P2P technology can play a powerful role for distributed self-management and control schemes in the power grids. In [18] is presented a conceptual model for managing sensitive information such as the personal health information. This proposal uses information from different health-care providers and it is supported by blockchain technology and P2P networks. The authors also do a security analysis for the proposed model. Data integrity is guaranteed, and the blockchain technology is used to offer an immutable of the data record. The authors state that the model presented a good experimental performance in term of data dissemination. On the other hand, several IoT data marketplaces have been deployed on P2P networks. For example, a review system is proposed in [17], which can confirm the reputation of a data owner or the data traded in the P2P data marketplace. This study is based on Etherum model and P2P networks, which is used to face the limitations of the client-server model such as security vulnerability or server administrator's malicious behavior. The integrity and immutability of the registered reviews are assured too.

3 P2P Background

A P2P network is an overlay network which is established through TCP (Transport Control Protocol) or HTTP (Hypertext Transfer Protocol) connections [14]. An example of a P2P network is shown in Fig. 1. Here, the participating nodes form the overlay P2P network on the top of the physical network [12].

P2P networks can be classified as unstructured and structured. Participating nodes in an unstructured P2P network are chosen randomly. This type of network is often used for distributed and heterogeneous systems [15], where it is not possible to maintain strict restrictions on the placement of control data and network topology. Unstructured P2P networks use flooding communication protocols which cause a large amount of network traffic [11]. Unstructured P2P networks can be further divided in centralized P2P, pure P2P and hybrid P2P. On the other hand, structured P2P networks maintain a close coupling between the network topology and the location of data using a hash table (DHT) [13]. This table is used to precisely define the data placement and lookup operations, and the DHT mechanism handles the peers joining/leavings on the overlay. Each peer has a routing table with information of links to a small subset of peers.

A computing system involves different computer resources, policy, management strategies and applications with a variety of computing requirements. The users of a system either as producers or consumers of resources have different objectives, strategies

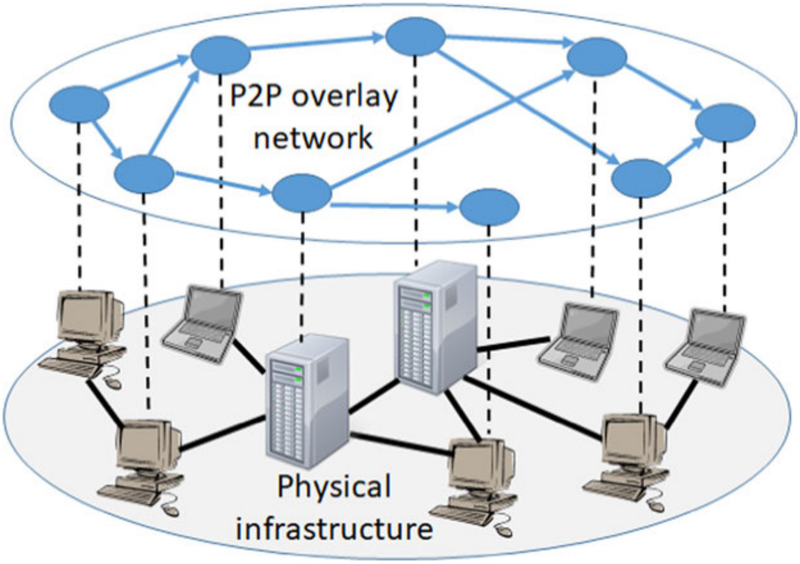


Fig. 1. A P2P network is an overlay network.

and demand patterns which cloud be considered during the design of the system. In most cases, users and resources are dispersed through the world. The management of these distributed resources is often complex because traditional approaches based on centralized schemes to optimize performance measures cannot be used. This work reviews some collaborative computing infrastructures to achieve better connectivity and efficiency in the dissemination of content. In some scenarios, collaboration occurs between nodes that work as peers (or equals) to distribute content, while in other cases sources collaborate with each other to distribute their workload. P2P networks introduces different benefits such as decentralization, cost reduction, resource aggregation, scalability, dynamism, fault resilience, self-organization and anonymity. Since the Napster advent, a significant number of P2P applications have been developed. P2P applications have become a large category, which have been categorized into major areas such as file sharing, distributed computing, collaboration, media streaming. In this work, a P2P collaborative architecture for content distribution and storage is proposed.

4 P2P Collaborative Architecture

Our proposal architecture is shown in Fig. 2. Our architecture is an extension of our works presented in [21] and [22]. This architecture has two levels. In the first level there is the main pool, which contain a defined number of trackers. In the second level there are different pools with peers and different function. For example, pool 1 can be used to storage and distribute video files, while pool 2 can be used to storage and distribute music files. Thus, we can assign different function to each pool. Nodes in each pool work as peers, and they collaborate with each other to store files, in such a way that a virtual storage space is created throughout the system. Trackers in the main pool can be coordinate with each other to distribute the different file types through the system.

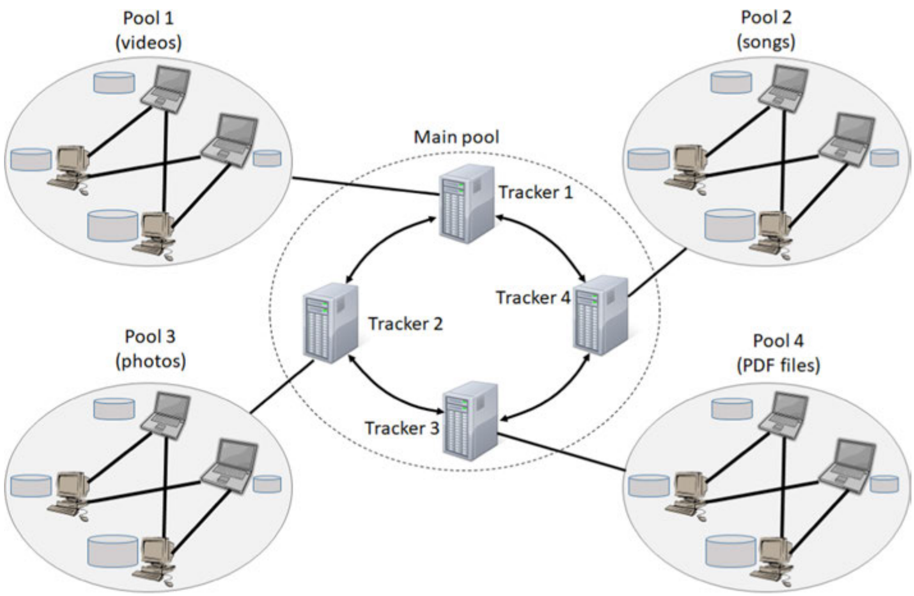


Fig. 2. Proposed collaborative P2P architecture.

Different transparency levels for the users can be considered in this architecture. In this way, operation details of the system are hidden from the user. P2P infrastructure is used by the system to support operations of storage, back up, and file synchronization and sharing. In addition, the trackers control parameters related to availability, reliability and quality of service of the peers in each pool. Therefore, these parameters are also transparent to the users. On the other way, the system can be initialized by any tracker in the main pool. In each pool, the communication between the peers is done via the tracker application, which manages the database where are registered the reports generated by the peers and their content list to be shared.

In our architecture the trackers are not storage servers, therefore they do not store files. Thus, when a file is received by the tracker from a peer, this file is redirects to another peer to be stored. Otherwise, when a peer wishes recover a file, it is requested by the tracker to peer where the file is stored. A tracker contain the database with information of each peer in the pool. This information is related to store capacity, dynamism and availability of each peer, which allow to define the reliability of the peer in the pool. In this scenario, all participating nodes are registered in the main server (tracker), which also monitors the behavior of each node in the system. The interaction between a tracker and a peer is shown in Fig. 3. In this case, each node in the P2P network runs a peer application, such that each node must receive and send files at the same time. To reach time goal, peer application performs both tasks simultaneously. The peer application also supervises each peer and reports its shared resources and behavior to the tracker in each pool. The peer application has a server part and a client part. The server part always is active to listen the request from the other peers, while the client part is responsible to upload files, display files and exit. Also, client part reports to the tracker all information related to the storage and upload capacity of its local computer in the pool.

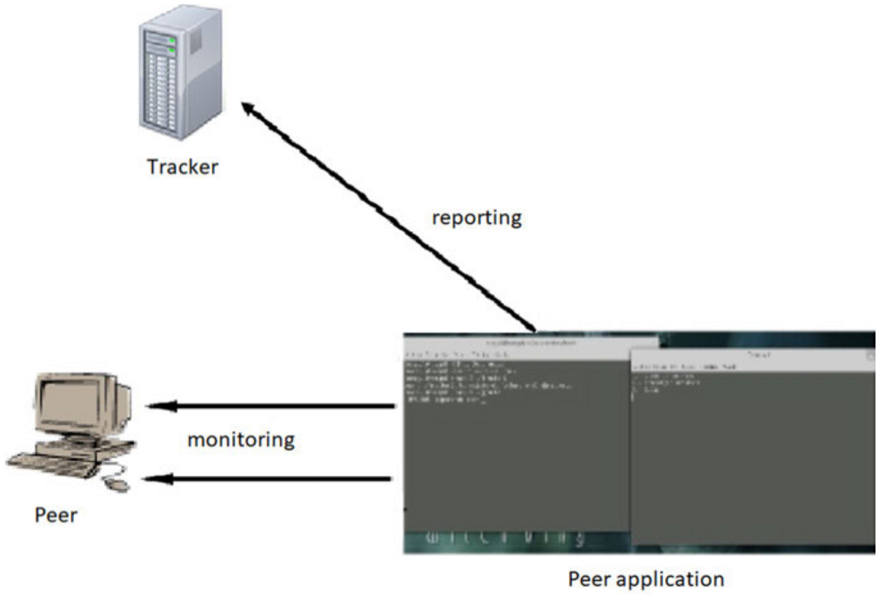


Fig. 3. Interaction between a peer and a tracker.

The tracker in each peer can classify the priority and importance of each file received and decides where (a specific peer) it should be stored. The tracker is responsible for establishing the communication between peers and routing the contents. Flow diagram of tracker operation is shown in Fig. 4. Each tracker has a database which has been designed to register and monitor information related to the peers. For each peer in the pool the following information is recorded in this database: physical address, IP address, date and time of the last connection, available space in disk, availability and dynamicity. To store content received from a sending peer, the tracker looks for host peers with similar level of reliability as the sending peer. The localization transparency is offered by the tracker to all users in the system. Thus, a sending peer does not know where is placed its content in the pool, because this activity is done by the tracker. When a peer wishes to recover a content, it should be requested through the tracker, which addresses the requested content from the host peer to the requesting peer.

Security in distributed and collaborative environments is a very important issue to be considered to reduce the risks to the information that is shared in different devices and physical places [23]. Therefore, there are various preventive, detective and corrective measures that must be considered to protect the information. Among the most important objectives that security services must meet are confidentiality, integrity, authentication, and non-repudiation. Cryptography enables these objectives to be achieved by allowing the meaning of the original messages to be altered through encryption and encoding. Cryptography can be symmetric and asymmetric (or public). The first method uses one key on both sides of the communication while the second method uses two keys, a different one on each side of the communication (private key and public key). The symmetric encryption algorithms perform their encryption by blocks of bits the size of the key, which is chosen randomly. Examples of symmetric encryption algorithms are DES, triple DES, RC2, RC4, AES [23].

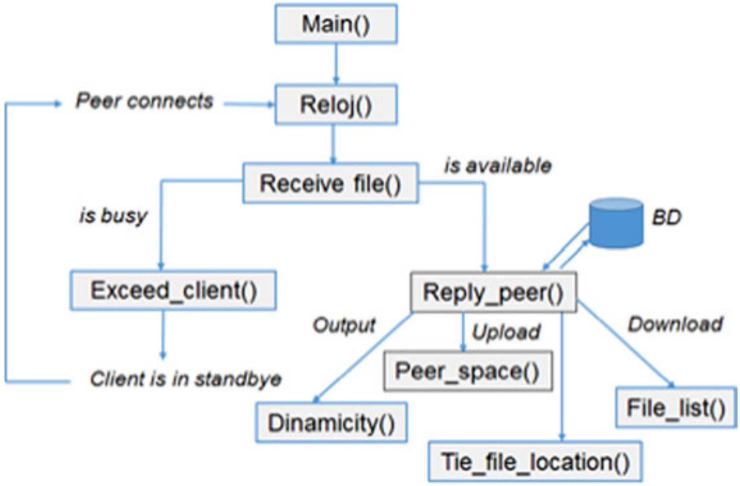


Fig. 4. Flow diagram of tracker operation.

Our proposal architecture considers a privacy and authentication framework to encrypt the files to be distributed and stored by the peers. Figure 5 shows how the security scheme interacts with the peer and tracker applications. There are a database with the passwords which are used to access to the system, and to store a file in a peer or to distribute a file from a peer to one or more peers. This database is managed by the tracker application in each pool. Each tracker also manages the authentication process. When a node arrives to the system it is authenticated by the tracker in each pool. Privacy is reached via encryption techniques. After a file is encrypted it will stored in a database located in a local peer. Our P2P collaborative architecture plans to use the AES encryption algorithm because it does not consume many resources when encrypting video files. This is very important, since this task can be done by each user on their own computers, so the computing capacities are variable. Therefore, if an algorithm is not efficient or consumes a lot of resources, it could cause problems with the equipment, such as making the encryption process take a long time.

The authentication process is done from point to point, that is, every time a user wants to enter the system, a message must be sent with their username and password so that it can be verified by the tracker with the database. If both data match then access is allowed. The password is stored in MD5 format by the tracker. Every time the password is authenticated, it is sent in MD5 format so as not to be visible to intruders. In the

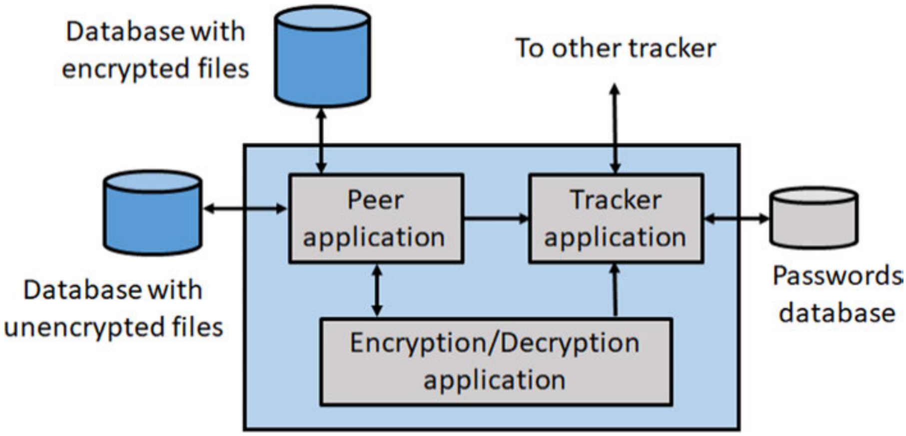


Fig. 5. Security framework.

encryption process, the user has to provide three different keys which are stored in the database, to be used later. In the encryption process, the encrypted files are stored in an attached folder of the software and its name is also stored in the database to be used when decrypting. When the files are transferred, they are sent encrypted, and together with this, a query is made to the database to know the encryption key of the file, which is sent to the user who downloaded the file for decryption. The data is managed through a database, which contains user data such as username, login password, and access permissions. In addition, the database also contains information about the files that users upload, which helps to make the search more efficient. These data are concentrated in the trackers, which helps to avoid losses in searches as normally happens in pure P2P systems where information is sometimes lost.

5 Evaluation

Our work is in progress. However, to evaluate our collaboration concept we have done an initial implementation of our proposed architecture. This prototype uses different entities such as entities are peers, trackers and databases, which are deployed on Linux using language C/C++. We used a reduced number of peers to compare the performance of a P2P network against a client-server network. In this case, 4 files of 28 MB are distributed between the same number of nodes for both networks, and the performance is measured in term of distribution time of the files to all nodes. Results are shown in Fig. 6.

We can see how nodes using the collaborative model can distribute the four files in a less time compared with the client-server model. This is because the client-server architecture needs to broadcast the four files to all nodes, while P2P model uses the collaboration between peers to distribute the four files to the all peers. Collaboration strategy reduce delivery time to all end-users.

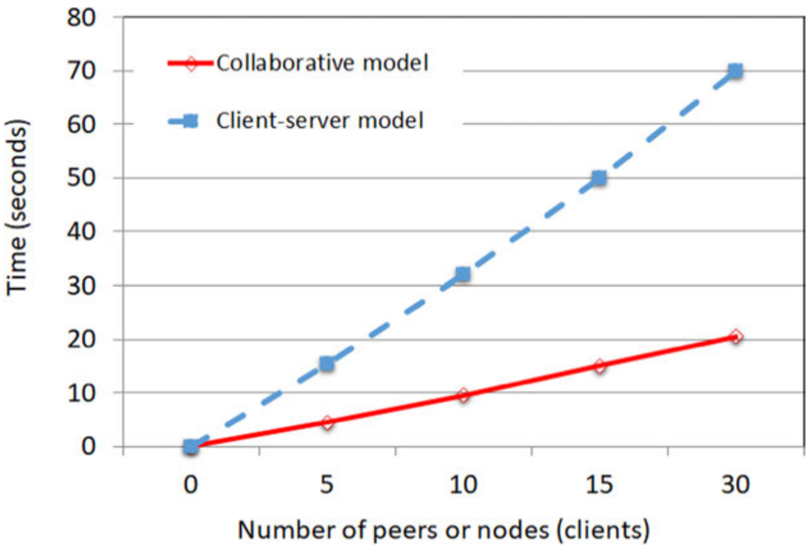


Fig. 6. Comparison of distribution time.

6 Conclusions

Content storage and distribution on Internet is very popular currently. In this paper, we present a P2P collaborative architecture for content distribution and storage management. Our proposed architecture integrate different strategies to face the different challenges for content distribution and storage such as authentication, privacy, quality of service, scalability and heterogeneity. We believe that collaborative P2P based on volunteer computing represent a good opportunity for small communities with common interests, because these paradigms represent a way to share resource and save money.

Our work is in progress, however initial results show benefits of collaboration compared with the traditional client-server model. Our work can be extended to different scenarios. For example, we can integrate in our architecture a coordination mechanism between the trackers, network coding or information dispersion techniques and its respective evaluations. Also, different security schemes can be studied in order to evaluate the encryption times and its impact in the file privacity.

References

1. Jung, Y., Lee, J., Kim, M.: Community computing model supporting community situation based strict cooperation and conflict resolution. In: Obermaisser, R., Nah, Y., Puschner, P., Rammig, F.J. (eds.) SEUS 2007. LNCS, vol. 4761, pp. 47–56. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-75664-4_6
2. Durrani, M.N., Shamsi, J.A.: Volunteer computing: requirements, challenges, and solutions. J. Netw. Comput. Appl. **39**, 369–380 (2014)
3. Bruque-Cámara, S., Moyano-Fuentes, J., Maqueira-Marín, J.M.: Supply chain integration through community cloud: effects on operational performance. J. Purchasing Supply Manag. **22**(2), 141–153 (2016)
4. Goyal, S.: Public vs private vs hybrid vs community-cloud computing: a critical review. Int. J. Comput. Netw. Inf. Secur. **6**(3), 20–29 (2014)
5. Nov, O., Anderson, D., Arazy, O.: Volunteer computing: a model of the factors determining contribution to community-based scientific research. In: Proceedings of the WWW 2010, Raleigh, NC, USA, pp. 741–750. ACM Press (2010)
6. Anderson, D.P., Fedak, G.: The computational and storage potential of volunteer computing. In: Proceedings of the 6th IEEE International Symposium on Cluster Computing and the Grid, Washington, DC, pp. 73–80. IEEE (2006)
7. Cunsolo, V.D., Distefano, S., Puliafito, A., Scarpa, M.: Volunteer computing and desktop cloud: the Cloud@Home paradigm. In: Proceedings of the 8th IEEE International Symposium on Network Computing and Applications, Cambridge, MA, USA, pp. 134–139. IEEE (2009)
8. Buyya, R., Murshed, M.: GridSim: a toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. Concurr. Comput. Pract. Exp. **14**(13–15), 1175–1220 (2002)
9. Androutsellis-Theotokis, S., Spinellis, D.: A survey of peer-to-peer content distribution technologies. ACM Comput. Surv. **36**(4), 335–371 (2004)
10. Milojicic, D.S., et al.: Peer-to-Peer Computing (Technical report HPL-2002-57R1). HP Labs, Palo Alto (2002)
11. Schollmeier, R.: A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In: Proceedings of the First International Conference on Peer-to-Peer Computing, Linköping, Sweden, pp. 101–102. IEEE (2001)
12. López-Fuentes, F.A.: Video Multicast in Peer-to-Peer Networks, Doctoral dissertation, Technische Universität München, Germany (2009)
13. Zheng, W., Liu, X., Shi, S., Hu, J., Dong, H.: Peer-to-peer: a technique perspective. In: Wu, J. (ed.) Handbook of Theoretical and Algorithmic Aspects of Ad Hoc, Sensor, and Peer-to-Peer Networks, pp. 591–616. Auerbach Publications, Boca Raton (2005)
14. Ripeanu, M., Foster, I., Iamnitchi, A., Rogers, A.: In search for simplicity: a self organizing multi-source multicast overlay. In: Proceedings of the 1st IEEE International Conference (SASO 2007), Boston, MA, USA, pp. 371–374. IEEE (2007)
15. Li, J., Vuong, S.: An efficient clustered architecture for P2P networks. In: 18th International Conference on Advanced Information Networking and Applications, Fukuoka, Japan, pp. 278–283 (2004)
16. Bandara, H.D., Jayasumana, A.P.: Collaborative applications over peer-to-peer systems—challenges and solutions. Peer-to-Peer Netw. Appl. **6**, 257–276 (2013)
17. Park, J.S., Youn, T.Y., Kim, H.B., Rhee, K.H., Shin, S.U.: Smart contract-based review system for an IoT data marketplace. Sensors **18**, 3577 (2018)
18. Rahmadika, S., Rhee, K.: Blockchain technology for providing an architecture model of decentralized personal health information. Int. J. Eng. Bus. Manag. (2018)

19. Marzal-Romeu, S., Salas-Puente, R., González-Medina, R., Figueres, E., Garcerá, G.: Peer-to-peer decentralized control structure for real time monitoring and control of microgrids. In: Proceedings of the 26th International Symposium on Industrial Electronics (ISIE), Edinburgh, UK, pp. 140–145. IEEE (2017)
20. Babaoglu, O., Marzolla, M., Tamburini, M.: Design and implementation of a P2P cloud system. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing (SAC 2012), Trento, Italy, pp. 412–417. ACM Press (2012)
21. López-Fuentes, F.A., García-Rodríguez, G.: Collaborative cloud computing based on P2P networks. In: Proceedings of the 4th International Workshop on Collaborative Enterprise Systems (COLLABES-2016) co-located with the IEEE-AINA, Crans-Montana, Switzerland (2016)
22. López-Fuentes, F.A., Orta-Cruz, C.A.: A secure P2P architecture for video distribution. In: Proceedings of the First International Workshop on Internet-Scale Multimedia Management (WISMM 2014), co-located with the ACM Multimedia Conference, Orlando, FL, USA (2014)
23. Stalling, W.: Network Security Essentials: Applications and Standards, 4th edn. Pearson (2011)
24. Varghese, B., Buyya, R.: Next generation cloud computing: new trends and research directions. Future Gener. Comput. Syst. **79**, 849–861 (2018)
25. Dubey, K., Shams, M.Y., Sharma, S.C., Alarifi, A., Amoon, M., Nasr, A.A.: A management system for servicing multiorganizations on community cloud model in secure cloud environment. IEEE Access **7**, 159535–159546 (2019)

Data Obfuscation in Network Coding to Mitigate the Effects of Pollution Attacks

Raúl Antonio Ortega-Vallejo and Francisco de Asís López-Fuentes(✉)

Department of Information Technology, Universidad Autónoma Metropolitana - Cuajimalpa, Av. Vasco de Quiroga 4871, Cuajimalpa, 05348 México City, México
raulantonio@protonmail.com, flopez@cua.uam.mx

Abstract. Network coding is a technique mainly used to maximize the throughput, minimize the delay, or optimize the reliability in the communication networks. However, network coding presents vulnerabilities problems in security terms and is susceptible to security attacks. We analyze impact of a security attack called pollution attack in traditional network coding based on butterfly scheme and propose a solution to deal with this problem. Cryptographic algorithms can protect the information into the data stream combined with network coding, but the processing cost could be high. Our solution considers a hybrid protocol which combines AES algorithm with obfuscation techniques. Our results show that our protocol has a better performance than fully encryption in term of processing time.

Keywords: Security · Network coding · Networking · Information theory

1 Introduction

According to the Cisco report [18], there are 5.3 billion users connected to the Internet. The consumption of data has increased, as has the demand for network infrastructures with higher performance (speed, throughput) [18]. In recent years, network coding has been studied as a technique from information theory to optimize the transmission of messages in communication networks [1]. Network Coding is done in the intermediate nodes which relay received messages toward somewhere destiny but adding other function about the simple forwarding of data [1]. Several benefits in the communication networks related to network coding has been related in the literature (e.g.: maximize the throughput, minimize the delay, optimize the reliability) [19].

For understand the essence of network coding, let us consider a network for a multicast scenario where the intermediate nodes compute algebraic operations on the packets received from the sources. This allows create packets with combined data, this means that a codified packet content more information than non-codified packet. The authors in [1, 2] use a butterfly topology to expose the communication procedure between nodes using network coding.

In Fig. 1a is shown a simple forwarding sending with multicast environment. The main source node S is sending the packets a and b, to nodes N1 and N2, which just are forwarding the streams a and b to node N3. The node N3 is a relay node and decides

M. F. Mata-Rivera et al. (Eds.): WITCOM 2022, CCIS 1659, pp. 1–17, 2022.
https://doi.org/10.1007/978-3-031-18082-8_1

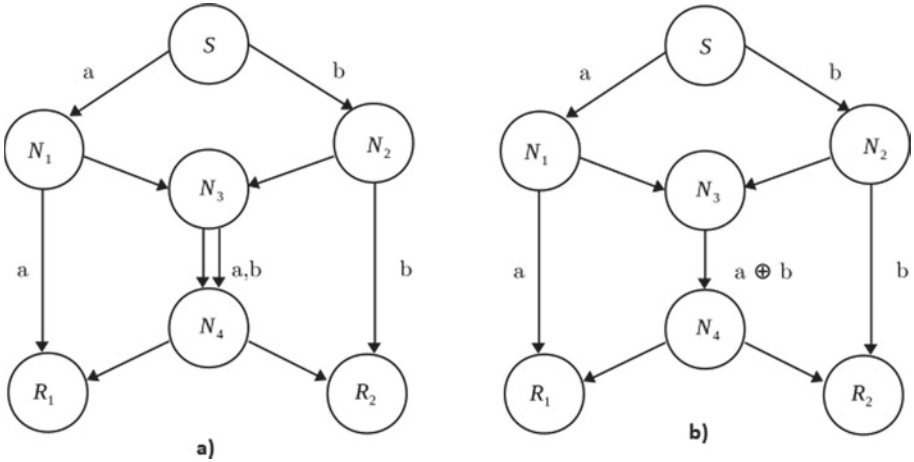


Fig. 1. Butterfly topologies: a) simple forwarding, b) network coding [1].

when forwarding the packet, *a* or *b* but not both to the same time, because it is not possible [2]. The arrows between the nodes N3 and N4, reveal the times required to complete the forwarding of packets (2 per unit of time) [1, 2]. On the other hand, in Fig. 1b the multicast with network coding improves the times to relaying the packets, because only takes 1 (per unit of time) for N3 to send a codified packet to N4 [1, 2].

In the traditional network coding, the intermediate node (N3 for this case, Fig. 1) computes an xor (⊗) from packets received [1]. In this case, in Fig. 1b, the packets *a* and *b* are combined as ($a \otimes b$) during the encoding process done by node N3, which sends the result to N4. This node is forwarding ($a \otimes b$) to the receivers (R1 and R2). The sink node R1, recovers both packets *a* and ($a \otimes b$), while the sink node R2 recuperates *b* and ($a \otimes b$). After that, using the decoding process, the receivers can know the sense of ($a \otimes b$) for each one. For example, the node R1 has the packet *a* from N1, computing in sequence another binary operation, $a \otimes (a \otimes b)$ gets the packet *b*. The similar way is for R2, which recovers the packet *a*, using $b \otimes (a \otimes b)$. This means that each sink node, cans recover the packets *a* and *b* to the same time (1 per unit of time) and the limit capacity of each communication channel is not exceeded [1–3].

With the advantage times, the paradigm network coding proves a better throughput compared to a simple forwarding method [2–4]. The investigation in [2], denotes that the latency problems could be reduced. A use case of network coding is when the file to be transmitted can be separated in multiple files. For example, if the topology network is required for steaming media like video and audio N1 could be the video source and N2 the audio source, harnessing the distribution of the butterfly topology. The authors in [4], introduce a framework to optimize a variant of this paradigm, such that approach is on CPU processing devices (nodes) to allow network members use light algorithms by encoding and decoding processes. In [5] is shown another application related to multimedia storage using peer to peer networks with network coding. The authors compared the typical server to client distribution centralized against decentralized scheme using codification.

However, this paradigm is susceptible to security attacks [6, 7], because it has not focused on security for interception attacks [6]. In this paper, we study the vulnerability consequence of the pollution attack in the traditional network coding scheme. Our motivation is based on investigations which ensure possible damage in the communication networks when a pollution attack occurs [6]. There are different network coding schemes [2, 3], in this work, we have chosen the traditional network coding to know how the sending times can be optimized.

The rest of this paper has been organized as follows. Pollution attack in traditional network coding is explained in Sect. 2. Related work is introduced in Sect. 3. We explain the steps used in our methodology in Sect. 4, where also are explained how the obfuscation techniques and network coding are implemented an integrated to mitigate a pollution attack. Then, evaluation of our proposal and its obtained results are discussed in Sect. 5. Finally, conclusions and future work are presented in Sect. 6.

2 Background

A pollution attack is when an attacker or process does a channel interception and sends its own stream instead of a legitimate flow [22, 23]. The buffer of attacker could contend rubbish data [6, 7] and the receiver could read the corrupt packets. For example, in Fig. 2 if an attacker intercepts the interaction of N2 and N3, the attacker does a stream to the node N3 instead of N2 flow, the node N3 could be an *xor* with rubbish data and a legitimate packet (*a* or *b*). A corrupted packet is the result of this, furthermore other nodes could expand the attack to the rest of the network. Figure 2 shows how N4 forwarding the corrupt packet, in consequence the sink nodes (R1 and R2) cannot recovery *a* or *b*.

The real damage impact of pollution attack is depending of the objective of the attacker, for example if the attack pretends generate a *DOS* (*Denial of Service*) from a pollution attack, perhaps the node N3 is the possible candidate to attack first. In this work, we simulated attacks at various points if the network using the traditional codification (XOR). This allows analyze the possibilities for the attacker and evaluates the damage on the network. As result, we propose a protocol which uses a cryptography algorithm and obfuscation techniques.

In computational context "obfuscation" refers to produce non-understandable text for humans [14]. The mean intention is hide information or procedures at the code of program or hide some strings from one text [14]. Obfuscation make a program more difficult to understand because its structure is changed, while the original functionalities is preserved [24]. Sometimes obfuscation activities are used for the cybercrime [12]. This is possible, because if a computational forensic scientific want to use a special software to investigate a suspicious code, maybe the software cannot recognize anything from the file. Furthermore, the forensics could be confused with the obfuscated code. However, we believe obfuscation can be used for positive activities such as protect valuable information [12, 13]. Our objective in this work is to develop a protocol which allows to mitigate the damage produced by pollution attacks. Our proposed protocol tries the attackers desist to steal the information from the packets.

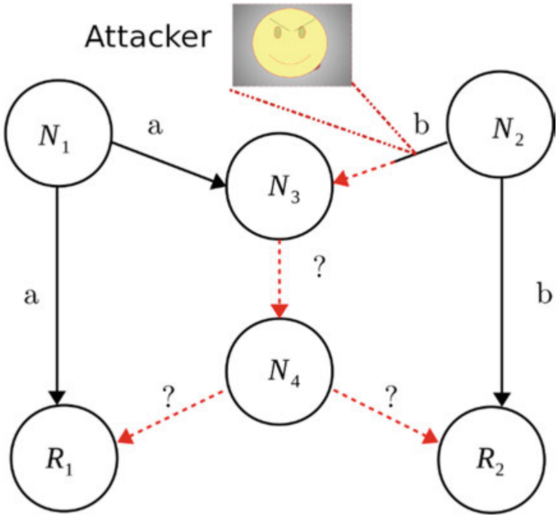


Fig. 2. A pollution attack example between N2 and N3 nodes.

3 Related Work

Recently security for network coding has generated great interest for the researchers [6, 8–10]. The adversary model in [6], explains about several ways network coding can be attacked. We found interesting works about it, with proposals, models and techniques using cryptography. There are contributions centered in cryptographic methods as the *KEPTE* model [8]. This model has a key distribution scheme for encrypt and decrypt the vectors. In this model there is a *KDC (key distribution center)*. For example, with this proposal, we have two special vectors as a public key and as a private key. For that reason, the nodes must interact with the KDC and request these keys to protect the flows. Unfortunately, this asymmetric model only has one generation of keys for encrypt and decrypt the source vectors [8]. In [9] authors explain the possible vulnerabilities that this model has. To integrate a key distribution with network coding is complicated. However, author in [9] proposes a dynamic generation of secret keys to complement the KEPTE approach.

Another model is *SPOC* (safe practical of network coding) [10], where the authors use end-to-end encryption for the data on network coding. The algorithm AES (Advanced encryption standard) is used for making it. One interest of using AES in SPOC, is because this scheme requires an encryption mechanism with ciphertext size equal to that of the plaintext. This means that the intermediate nodes can operates on the locked coefficients without the need for the decryption keys [10].

The authors in [20] present a scheme for detect suspicious activities or attacks using network coding. The idea is to has a SDN (centralized software defined controller), which seem a special tunnel for classify packets [20]. On the other hand, the SDN takes the byzantine problem [6]. Where the node, makes reports about the packet, but maybe this will produce more connections than we need. Anyway, the focus in [20] was on mobile technologies.

In the obfuscation context, the authors propose in [11] to use obfuscation for network coding. This model called *CNC* (*Conceal network coding*), does obfuscation using bitwise movements during the codifying process [11]. Other proposal for obfuscate data for this paradigm is [16], where exists an interesting comparison between typical and homomorphic encryption algorithms. Where the homomorphic strategy is more expensive for the nodes. The authors use *SRC* (*Secret random checksum*) for each node that receives a packet. This allows to the receivers know which packet was corrupted, furthermore is possible to discard it. Also, in this approach an attacker cannot recover the file completely if the vector file was obfuscated [16]. For example, if an attacker intercepts a fragment from the original file, means the attacker only got a file portion which could be an was obfuscated part and it is not enough to recovery the entire file. Considering this characteristic, our investigation tries to develop a strategy to confuse the attacker to intercept the obfuscated packets during a pollution attacks instead of the legitimate packets.

4 Methodology

Contrasting with the CNC proposal [11], our obfuscation protocol is used in the sources and receivers. Therefore, we do not append more processing work to the intermediate nodes. Furthermore, these nodes should not know the content of the packets, only parameters as the information of its destiny. In this way, we avoid possible user privacy affectation in the intermediate nodes. The source nodes have the task of protect the flows before to send it, while the sink nodes know the mechanism for recovery the information from the packets. Thus, in the intermediate nodes only network coding operations are implemented. To create the butterfly topology (see Fig. 1a), we classified the groups for each node. In this case, we have the *source nodes* (N1 and N2), the *intermediate nodes* (N3 and N4) and *sinks nodes* (R1 and R2), such that we have 6 nodes. We decided to employ our lab simulated occupying the same machine for set up the network. It was possible with Linux containers, where everyone represented a node. This allows to prepare a specific network configuration for some nodes. For example, the IP address and port number. In next step, we set up the specifications about the files related to their size and type. Because on an Ethernet real scenario there are few limitations to transfer files, we do not specific in our program any file type to be transmitted on the Ethernet network. However, in our experiments we have used text, images, binaries, and pdf files. The source nodes do a vector representation using a serialization process. The network can operate in three modes. 1. Using simple codification. 2. Encrypting the source vectors. 3. Using our proposed protocol. When the network uses the first mode, a file is serialized, and the vector without protection is transmitted over the network. In the second mode, a file is serialized, and the vector is encrypted using AES algorithm (with 192-bit keys). For example, if file A is serialized and encrypted, such that | A | is equal to | AES (A) |. This is the encrypted vector that would be sent, merged, and retrieved. The sinks nodes known the secret key to decrypt it. On the other hand, the third mode use the obfuscated technique with network coding. In this mode the obfuscation procedure is done in the source nodes. Figure 3 shows the output final vector to transmit with the treated packet using our proposed protocol.

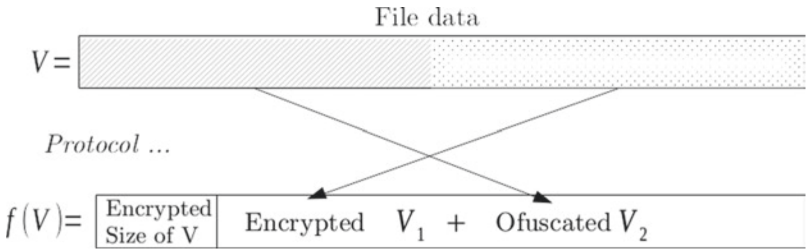


Fig. 3. Final vector as output of our proposed protocol.

In Fig. 3, V represents the vector which is serialized by the protocol. V is split in two sub-vectors (V1 and V2), such that |V| = |V1| + |V2|. The sub-vector V2, is obfuscated using arithmetic operations. For example, V2 contains the string “ABC”, in binary the value is “01000001 01000010 01000011”. Thus, we can turn it to “CBA”, so the binary file is “01000011 01000011 01000001”. Applying some changes, we have “1011110 10111101 10111100”, that is the result of interchange the 1 and 0. This produces the ascii value “^½¼”, which is very different to “ABC”. This allows to generate operations which can be included in our protocol, so that we can know the changes that were made to the values of V2. Once the operations with V2 are finished, we can concatenate V1 and V2 to form an obfuscated vector, which may be more difficult to be interpreted by an attacker. Figure 3 shows the result using the third mode (vector obfuscation). The operator ‘ +’ represents a concatenation on the final vector (buffer). The “Encrypted size” label indicates the size of the obfuscated vector. This information helps to the receiver nodes know the dimension of the vector. We assume that the source nodes and the sink nodes know the secret key and length for crypt and decrypt vectors (AES of 192-bit). Note that the encryption algorithm is used AES192(|V|/2) times, this mean its cost was reduced to the half of the vector. We believe that it is not necessary to encrypt the entire vector to protect the information of the nodes.

Once the source nodes have processed the vector under our protocol, to recover an encoded vector must be done the reverse process. For example, we know that the vector is composed of two sub-vectors, then we must swap the positions of the data blocks. Thus, if in Fig. 3, the sub-vector V2 is placed at the beginning, it must swap its position with V1. In other words, the vector must be rotated. After that, we need to interpret the sub-vectors, because one has been encrypted (V1) and the other has been obfuscated (V2).

Our methodology has the following steps:

a) First, we implemented the traditional network coding scheme. The network should support any extension of file. However, we use images files as input vectors.
b) Then, we simulated different attacks to study the impact in all nodes included the sinks nodes. This is important because we can know where an attack causes the most damage, if our intention is avoiding that damage be expanded. Our study with different cases of pollution attacks and its methodology were presented in [21].

c) We use a protocol based on the AES algorithm, but it is not used to encrypt all data. The length of key for AES is 192-bit, and both sinks nodes as sources nodes know the key to decrypt the encrypted vector.
d) After that, the AES algorithm is used to protect against the interception attacks, because it could limit the chance for the attacker.

Finally, we compare the overall transmission times using only traditional network coding with network coding using AES (192-bit) and with our protocol (AES of 192-bit + Obfuscation).

5 Evaluation and Results

In this project, we develop our prototype using the C language of programing, and the GPG-Libcrypt library for the cryptography algorithms. Our experiments were done in a computer powered by a 2.16 Ghz Intel Celeron processor and 8GB of RAM. We use Linux as our operating system. Initially, we tested our prototype with text files as input vectors for check communication between the nodes. However, our network coding prototype supports any extension of file. Through Linux containers we can specify the network options for each node of the butterfly network. Table 1 describes the necessary configurations for each node.

Table 1. Network configuration nodes.

Node sets	Description	IP Address
N1, N2	Sources	172.18.0.2-3
N3, N4	Intermediate/Relays	172.18.0.4-5
R1, R2	Sinks	172.18.0.7-8

Our evaluation is divided in two parts. In the first part, we evaluate de network coding performance and the impact of pollution attack in the network coding operation. In the second part, we evaluate the network coding performance using two different security strategies.

To evaluate the first part, we tested a traditional network coding based on the butterfly scheme. The sources N1 and N2 distributes two different image files called “photo1.jpg” and “photo2.jpg”. Figure 4 shows the activities monitor running in node N1, where a JPG file named “photo1.jpg” is processed. In this example, the file name is the same as the name of the image inside the file system of container where node N1 is hosted. This node is configured with the IP address 172.18.0.2, processes the file and serializes it to be transmitted to nodes N3 and R1.

In Fig. 4, we can see that our program takes the time of each file processing procedure and the time of interaction with the nodes. This information allows to feed a log file, with the activities of source node N1. All nodes of our network record their binnacle in a log file. These log files contain data such as time and connections between the nodes on the network. With these records, we can identify possible time variations during the communication between the nodes. Figure 5 shows a scenario similarly to previously described, but for source node N2.

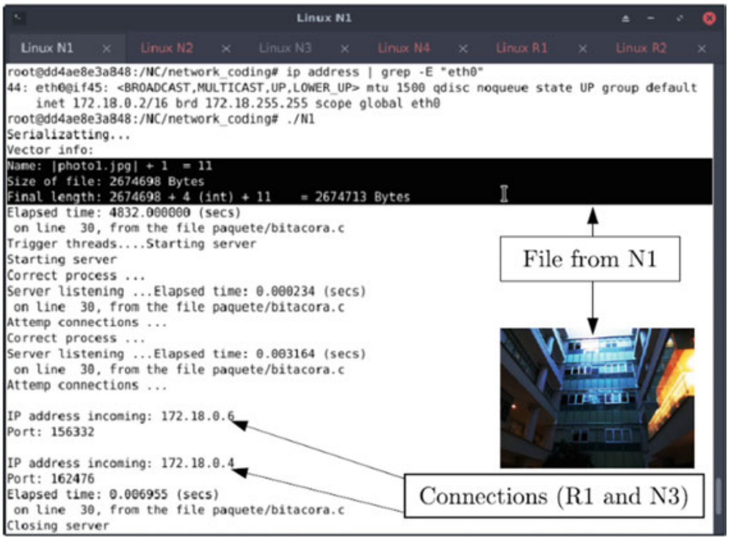


Fig. 4. Monitor of N1 sending the file "photo1.jpg".

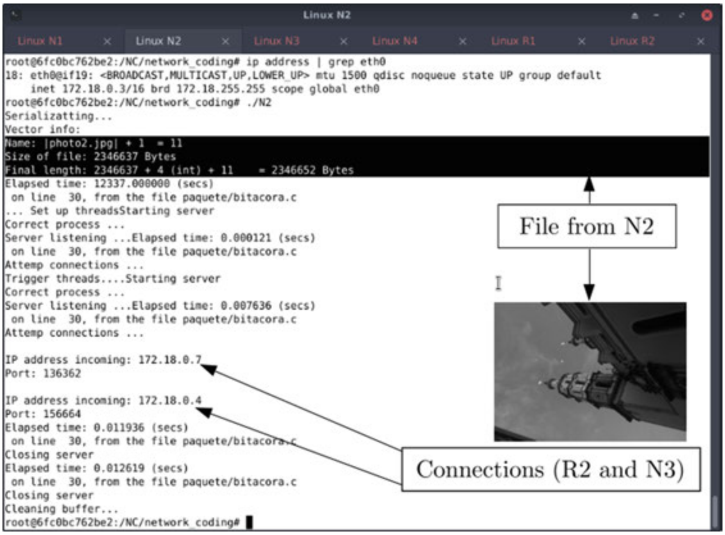


Fig. 5. Monitor of N2 sending the file "photo2.jpg".

Figure 6 shows the monitor for node N3, which is the intermediate node responsible to apply network coding in both encrypted vectors received from source nodes N1 and

N2. It is important to note that both vectors must have equal length to be encoded. Then, the encoded packets are sent to node N4. If the received vectors in the node N3 are of different size network coding is not done, and node N3 simply forwards the incoming packets to node N4.

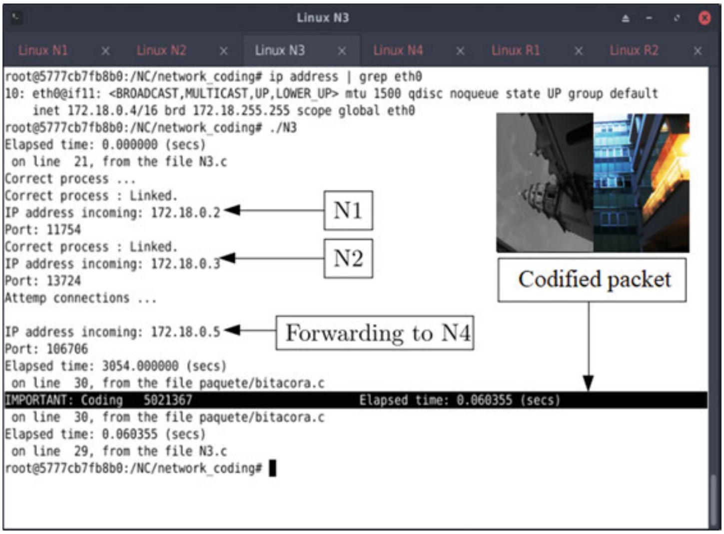


Fig. 6. Monitor for node N3 coding the packets from N1 and N2.

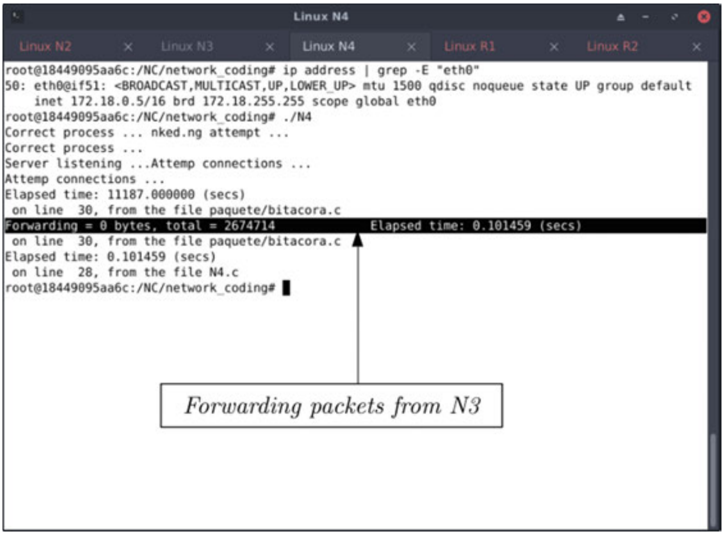


Fig. 7. Monitor for node N4 forwarding the flows received from node N3.

On the other hand, Fig. 7 shows the processing performed by node N4, which has the IP address "172.18.0.5" and receives the flow from node N3. Consequently, node N4 broadcasts the encoded packets to the sink nodes, in this case to the nodes R1 and R2.

The receiving node R1 receives the files "photo1.jpg" and "photo2.jpg" at the same time. We can see this scenario in Fig. 8. It is necessary to specify that the encoded packet sent by node N4 must be treated under the decoding process. We can see that our program retrieves the original file name and it is recorded in the registers within the log file in node R1. Figure 9 shows the monitor for node R2 receiving both files at same time. Node R2 proceeds same than node R1 to recover both files.

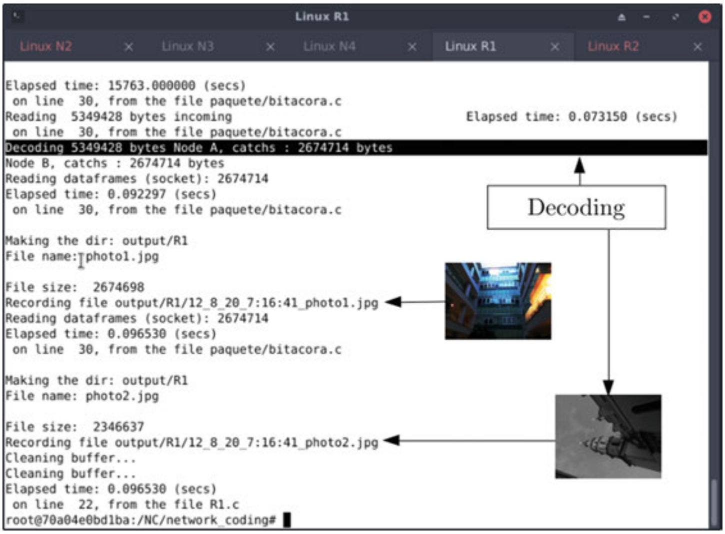


Fig. 8. Monitor for node R1 receiving both files at the same time.

Continuing with our methodology, the next step is to simulate stream pollution attacks. We simulate an attacker who intercepts the communication from node N2 and N3. Figure 10 shows a scenario for a pollution attack. For this case, N2 multicasts an image to sink node R1 and to intermediate node N3. Node R1 receives the correct packet, but node N3 does not receive the correct packet because it has been intercepted by an attacker. Then, the attacker can do a pollution attack, and the vector image presents errors or rubbish data. Thus, an image presenting damages has been saved in node N3. This case occurs if the attacker injects extra or rubbish data into the legitimate buffer.

The effects of this pollution attack in the node R2 is shown in Fig. 11. Node R2 is a receiving node (or sink node), and we can see how it can recovery only image "photo1.jpg" because imagen "photo2.jpg" sent from node N2 has been corrupted in the intermediate node N3. In this way, we can see how a pollution attack can corrupt a file.

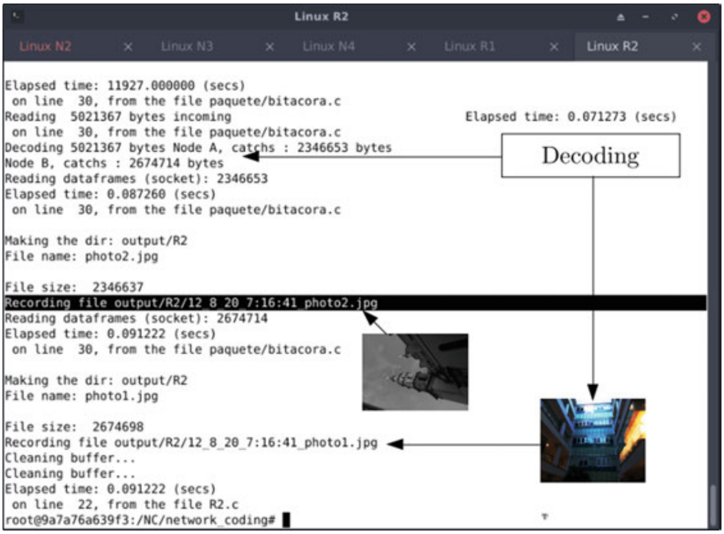


Fig. 9. Monitor for node R2 receiving both files at the same time.

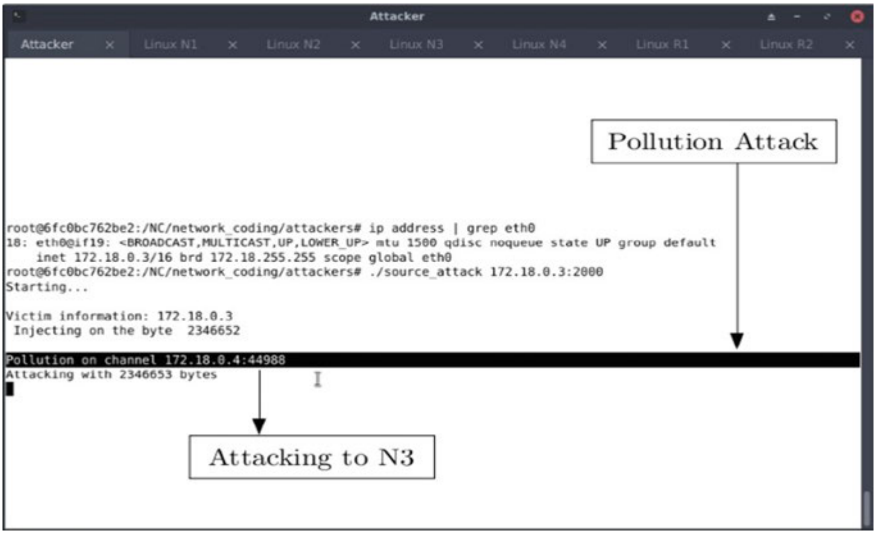


Fig. 10. An attacker intercepts the message between source node N2 and intermediate node N3, and a pollution attack can be done.

Another possible case is when the image could not be recovery in its totality (incomplete imagen). For example, we simulate an intercepted communication between the source node N1 and the sink node R1. This case is shown in Fig. 12. The attacker alters the file “photo1.jpg” sent to node R1 from node N1. Therefore, the file “photo2.jpg” sent from source node N2 is unrecoverable because the file “photo1.jpg” is corrupted, and it cannot be used to decode the file “photo2.jpg” using network coding in this scheme. Figure 13 shows both files received in the sink node R1. We can see how both files have been affected by this pollution attack.

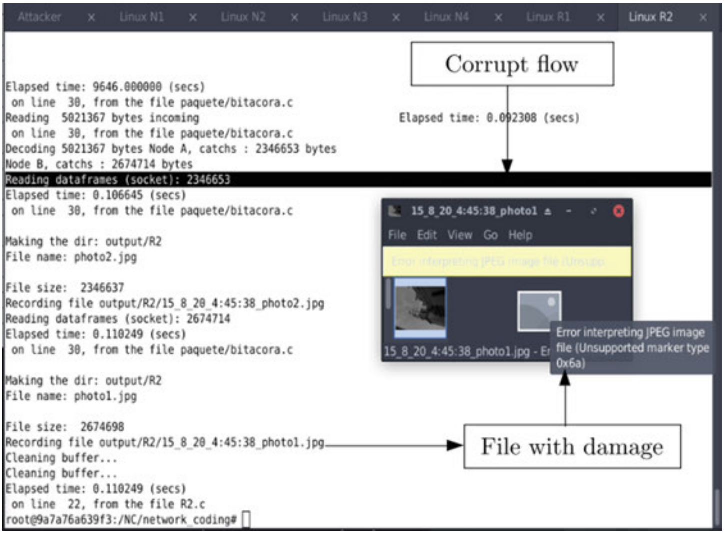


Fig. 11. Pollution attack effect on node R2. File from N1 is unable to recovery.

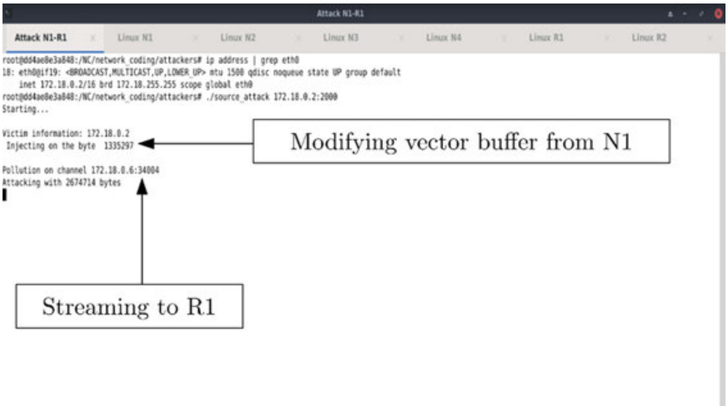


Fig. 12. A pollution attack is produced in the link between source node N1 and sink node R1.

We have done several experiments simulating other pollution attacks in different nodes or links of our butterfly network. In Table 2 we have summarized these potential pollution attacks and indicated how a specific encoded file is affected by a pollution attack.

Fig. 13. Affected files in sink node R1.

For example, if the attack is done in the link between source node N1 and sink node R1, only this receiver is affected with a corrupt packet. Other case is when the attacker intercepts all communication of source node N2 such that both sinks nodes receive incomplete packets from the sources. The worst case is when the packets in the intermediate node N3 is intercepted and the attacker cans control all communications with this node. This case produces a very negative impact on the network because each codified packet by node N3 is a corrupted packet. This case is shown in Table 2 too.

Table 2. Impact under pollution attack according to node interaction.

Intercepted	Sink R1	Sink R2
Link N1 to R1	Complete file	Corrupt file
Node N2	Incomplete file	Incomplete file
Node N3	Corrupt file	Corrupt file

The second part of our evaluation is developed once we know the negative impact of the pollution attacks can have on traditional network coding (butterfly scheme). Now, we need to consider different security strategies to mitigate the pollution attack impact. In the first strategy the source vectors are fully encrypted. Here, source and sink nodes know the secret key to encrypt and to decrypt the data. This strategy implies a higher cost of computing power, which we try to reduce in our second strategy that uses our protocol based on the obfuscation techniques that we have explained in Sect. 4.

The results from both strategies are compared with the results obtained using network coding only. Figure 14 shows results from our experiments in term of average time required by the computational processing for each strategy. Results in Fig. 14 are obtained by processing and transmitting a file of 5MB from two source nodes to two sink nodes.

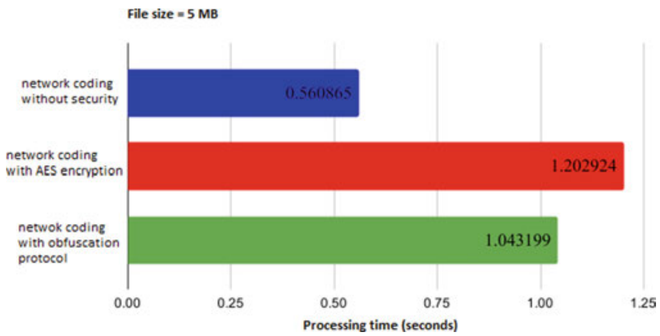


Fig. 14. Experiment with file size of 5 MB.

The elapsed time considers encryption time of original file in the source nodes, obfuscate time (it only apply in the second strategy), network coding time in the intermediate node, decoded and decryption time in the sink nodes, and the transmission time in all nodes. In Fig. 14 we can see how results using our obfuscation protocol has a less processing time compared with fully encryption strategy in around 0.159725 s. Processing time using our obfuscation protocol is 86% greater than processing time using traditional network coding without data protection, but processing time using our protocol is 13.3% less than network coding using fully encryption.

To experiment our protocol with files of different sizes, we have tested used files of 15 MB and 30 MB. Results for processing and transmitting the file of 15 MB using the three strategies (network coding without security, network coding with our obfuscation protocol and network coding with fully encryption) are shown in Fig. 15.

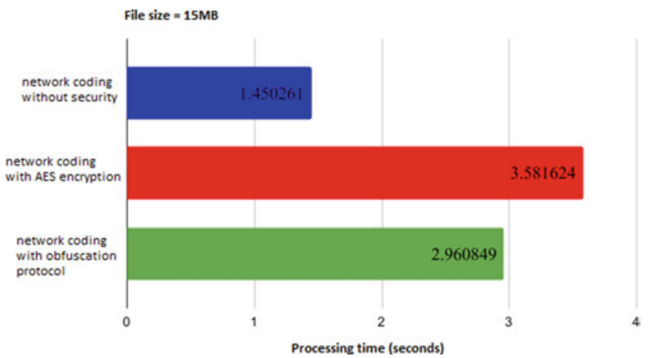


Fig. 15. Experiment with file size of 15 MB.

Experiments with a file of 30 MB are shown in Fig. 16. We can see in both figures that the difference between the use of fully encryption and our proposed protocol is more remarkable. For example, processing time using our protocol is less 17.3% than fully encryption for the file of 15 MB, while for the file of 30 MB this processing time is reduced in around 21.6%. Therefore, we see how fully encrypting the files is more expensive than using our obfuscation protocol. Finally, a general comparison is shown in Fig. 17. Here we can note how the difference in processing times between full encryption and our protocol increase as the processed files are larger.

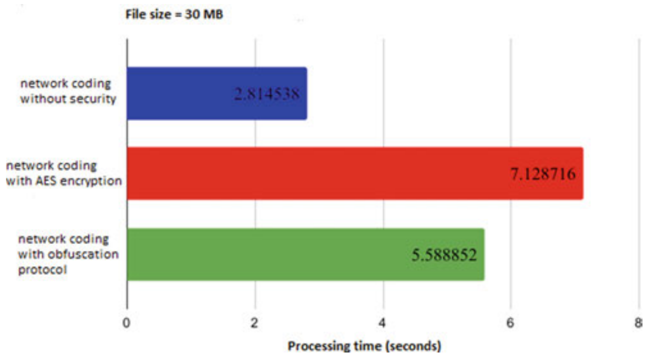


Fig. 16. Experiment with file size of 30 MB.

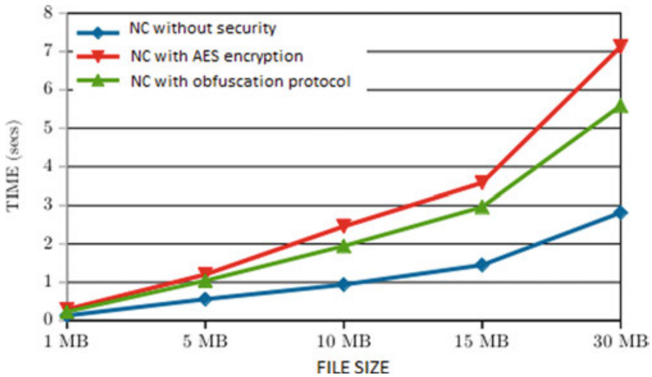


Fig. 17. Comparison of different file sizes from nodes N1 and N2.

6 Conclusions

Network coding has become an effective solution to increase throughput in the communication systems during last years. However, this technique is susceptible to attacks which can influence in the loss of data, and cause damage to the integrity of a communication system. In this paper we have shown how networking schemes based on network coding can be affected by pollution attacks. Data can be protected using some cryptography techniques. However, cryptography algorithms can have high processing costs.

Our protocol presented in this work integrates obfuscation techniques and cryptographic algorithms that allow files to be obfuscated before applying network encryption to them. Our results show that obfuscation combined with partial cryptography can be an efficient alternative solution to do secure network coding in communication networks because the processing times are more reduced respect to fully encryption. We believe that our proposal to face the pollution attacks in network coding open the doors to different possible solutions in the cybersecurity context. Our work can be extended in different directions. For example, a study and analysis about the percentage of message to be obfuscated o encrypted can be done. This study can help us to determinate the tradeoff between obfuscation and encryption for specific messages. Also, we can evaluate networking schemes different to butterfly where we can apply network coding with our proposed protocol.

References

1. Ahlswede, R., Cai, N., Li. S.-Y., Yeung, R.W.: Network information flow. IEEE Trans. Inf. Theory **46**, 1204–1216 (2000)
2. Yeung, R.H.: Information Theory and Network Coding. Springer, New York (2008). https://doi.org/10.1007/978-0-387-79234-7
3. Ho, T., Lun, D.: Network Coding: An Introduction. Cambridge University Press, Cambridge (2008)
4. Lucani, D., et al.: Fulcrum network codes: a code for fluid allocation of complexity. CoRR, vol. abs/1404.6620 (2015)
5. Lopez-Fuentes, F.A., Cabrera Medina, C.: Network coding for streaming video over P2P networks. In: IEEE International Symposium on Multimedia, pp. 329–332 (2013)
6. Zhang, P., Lin, C.: Security in Network Coding, 1st edn. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31083-1
7. Wu, X., Xu, Y., Yuen, C., Xiang, L.: A tag encoding scheme against pollution attack to linear network coding. IEEE Trans. Parallel Distrib. Syst. **31**(11), 33–42 (2020)
8. Xiaohu, W., Yinlong, X., Chau, Y., Liping, X.: A tag encoding scheme against pollution attack to linear network coding. IEEE Trans. Parallel Distrib. Syst. **25**(1), 33–42 (2014)
9. Liu, G.: Security analysis and improvement of a tag encoding authentication scheme for network coding. Wuhan Univ. J. Nat. Sci. **21**(5), 394–398 (2016). https://doi.org/10.1007/s11859-016-1186-1
10. Vilela, J, Lima, L., Barros, J.: Lightweight security for network coding. In: IEEE ICC 2008, pp. 1750–1754 (2008)
11. Hessler, A., Kakumaru, T., Perrey, H., Westhoff, D.: Data obfuscation with network coding". Comput. Commun. **35**(1), 48–61 (2012)
12. Alazab, M., Venkatraman, S., Watters, P., Alazab, M., Alazab, A.: Cybercrime: the case of obfuscated malware. In: Georgiadis, C.K., Jahankhani, H., Pimenidis, E., Bashroush, R., Al-Nemrat, A. (eds.) ICGS3 2011. LNICST, vol. 99, pp. 204–211. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33448-1_28
13. Bakken, D.E., Rarameswaran, R., Blough, D.M., Franz, A.A., Palmer, T.J.: Data obfuscation: anonymity and desensitization of usable data sets. IEEE Secur. Priv. **2**(6), 34–41 (2004)
14. Dalai, A.K., Das, S.S., Jena, S.K.: A code obfuscation technique to prevent reverse engineering. In: International Conference on Wireless Communications, Signal Processing and Networking, pp. 828–832 (2017)
15. Lin, D., Hamdi, M., Muppala, J.: Distributed packet buffers for high-bandwidth switches and routers. IEEE Trans. Parallel Distrib. Syst. **23**(7), 1178–1192 (2012)

16. Cascella, R.G., Caoy, Z., Gerlay, M., Crispo, B., Battiti, R.: Weak data secrecy via obfuscation in network coding based content distribution. In: 1st IFIP Wireless Days, pp. 1–5 (2008)
17. Liu, Q., Zhang, W., Ding, S., Li, H., Wang, Y.: Novel secure group data exchange protocol in smart home with physical layer network coding. Sensors **20**, 1138, pp. 1–17 (2020)
18. CISCO, Cisco Annual Internet Report, 2018–2023 white paper. https://www.cisco.com/c/en/us/solutions/executive-perspectives/annual-internet-report/infographic-c82-741491.html
19. Fragouli, C.: Network coding: beyond throughput benefits. Proc. IEEE **99**(3), 461–471 (2011)
20. Adat, V., Parsamehr, R., Politis, I., Tselios, C., Kotsopoulos, S.: Malicious user identification scheme for network coding enabled small cell environment. In: IEEE International Conference on Communications (ICC 2020), Dublin, Ireland, pp. 1–6 (2020)
21. Ortega-Vallejo, R.A., de Asís López Fuentes, F.: Vulnerability of network coding under pollution attacks. In: Vasant, P., Litvinchev, I., Marmolejo-Saucedo, J.A., Rodriguez-Aguilar, R., Martinez-Rios, F. (eds.) Data Analysis and Optimization for Engineering and Computing Problems. EAISICC, pp. 243–253. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-48149-0_18
22. Fiandrotti, A., Gaeta, R., Grangetto, M.: Simple countermeasures to mitigate the effect of pollution attack in network coding-based peer-to-peer live streaming. IEEE Trans. Multimedia **17**(4), 562–573 (2015)
23. Liang, J., Kumar, J., Xi, Y., Ross, K.W.: Pollution in P2P file sharing systems. In: IEEE INFOCOM 2005, vol. 2, pp. 1174–1185 (2005)
24. Behera, C.K., Bhaskari, D.L.: Different obfuscation techniques for code protection. Procedia Comput. Sci. **70**, 757–763 (2015)

2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) | 978-1-6654-6316-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/IEMCON56893.2022.9946543

Didactic Tool for Teaching Quality of Service Algorithms in Communication Networks

Francisco de Asis Lopez-Fuentes
Department of Information Technology
Universidad Autonoma Metropolitana-Unidad Cuajimalpa
Mexico City, Mexico
flopez@cua.uam.mx

***Abstract*— Nowadays, information technology opens great opportunities to transform teaching and learning. Currently, different platforms have been deployed to offer teaching services. These platforms can be built from computer systems that collect and analyze data, to those systems that interact with users and allow them to learn from these data. For example, a system can build animations to visualize and understand the operation of an industrial process or an algorithm. This paper presents a digital tool to support the teaching and learning of quality of service (QoS) algorithms in the communication networks. Quality of service is an important topic in the computers networks courses due to the current high demand for quality multimedia content on the internet. Our digital teaching tool tries to support the teaching/learning processes in this important communication networks topic.**

Keywords—Didactic software, quality of service, computer networks, simulation

I. Introduction

Today the information and communication technology (ICT) has had an unprecedented impact on society, culture, and the economy. ICT has broken into the way to teach courses at university or technical school, through digital tools which allow us to use interactive teaching material. Digitization turns out to be very important, since knowledge objects such as books, encyclopedias, or teaching resources are often represented as digital content. ICT has altered globally and significantly people's ways of communicating, entertaining, working, negotiating, governing, and socializing. “Knowledge multiplies faster than ever before and is distributed almost instantaneously” [1]. The impact of ICT increases productivity in various sectors of business activity, as well as creation of economies based on knowledge and innovation. The Internet has triggered a social revolution because of the way people interact with each other and with the rest of the world [2]. ICT can also support learning and the social construction of knowledge. ICT has a tremendous impact on education [8] because new ways of teaching and learning have allowed to students the acquisition and absorption of knowledge.

This paper presents a digital tool, which can be used for computer networks courses as a didactic support material. Specifically, our tool tries to explain the operation of the quality of service (QoS) mechanism used in the communication networks. In the context of communication networks, ITU [3] defines the quality of service as “the totality of characteristics of a telecommunications service that bear on its ability to satisfy stated and implied needs of the user of the service”. The primary goals of QoS cover dedicated bandwidth, controlled jitter, and lower latency (required by some interactive and real-time applications). QoS technology provides the elemental building blocks that will be used for future business applications, WAN (wide area networks) and service provider networks. In communication networks, QoS represents a set of technologies for guaranteeing the transmission of a given amount of data during a given time under certain quality metrics. QoS is an important requirement in IP networks, especially for sensitive applications such as video and voice transmission. Our didactic tool has two goals. The first goal focus on providing a teaching tool to the instructors which allows them to explain how QoS protocols operate in the communication networks. The second goal is to give students an opportunity to self-learn about the QoS protocols using our digital tool.

The rest of this work has the following organization. In section II, concepts about quality of service are presented. Section III introduces main quality of service techniques. Didactic tool is described in section IV. The article concludes in Section V.

II. Background

A heterogeneous communication network can provide the three basic levels of end-to-end QoS, which are [4]:

- *Best-effort service*. It is also known as absence of QoS. In this level, the best effort service is basically connectivity without any guarantee.
- *Differentiated service*. This case is also known as Soft Quality of Service (Soft QoS), and some traffic is treated better than the rest (fast handling, more bandwidth on average, lower loss ratio on average).
- *Guaranteed service (also called hard QoS).* This level does an absolute reservation of network resources for specific traffic.

The elements of the network must be able to handle the traffic overflow using a queuing algorithm to order the traffic, and then determine some priority methods. The main queuing algorithms to manage congestion and ensure quality of service are the following [5], [6]:

- First in-first out (FIFO)
- Priority Queuing (PQ)
- Custom Queuing (CQ)

- Weighted Random Early Detection (WRED)

Each queuing algorithm has been designed to solve a traffic problem and has a specific effect in the network performance. Descriptions of these techniques are given below.

III. QoS ALGORITHMS

This section explains the different QoS algorithms used to manage congestion and achieve acceptable QoS metrics in communication networks [4], [7].

A. First in- first out (FIFO)

This algorithm represents the simplest way to introduce quality of service (QoS) in the communication networks. FIFO algorithm stores the packets when the network is congested. After, the packets are sent across the network on the order in which they arrived. In some cases, FIFO is the default queuing algorithm. This algorithm does not require any configuration, but it has several limitations. More importantly, FIFO does not make any decisions about packet priority, the order of arrival of the packets determines its location in the buffer. The algorithm also does not provide protection against applications with irregular behavior originated from their sources. Explosive sources can cause large delays in the delivery of time-sensitive application traffic due to the network traffic generate. FIFO queuing was the first step to control the network traffic, but todays intelligent networks need more sophisticated algorithms.

B. Priority Queuing (PQ)

Priority queuing (PQ) ensures that important traffic gets the fastest handling at every point where it is used. This algorithm was designed to give high priority to the most important traffic. This queuing algorithm can relax the priority according to the network protocols (for example IP or AppleTalk), the input interface, the packet size, or the destination/source address. In priority queuing, each packet is placed in one of the four queues, which are defined based on an assigned priority. This priority can be high, medium, normal, or low. Figure 1 shows the flow diagram of operations of this algorithm.

In this algorithm, the packets that are not classified by the priority mechanism fall into the normal queue. During transmission, this algorithm gives an absolute preferential treatment to the highest priority queues over the lowest priority queues. Priority queuing is useful to ensure that mission critical traffic traverses multiple WAN links with priority hiring.

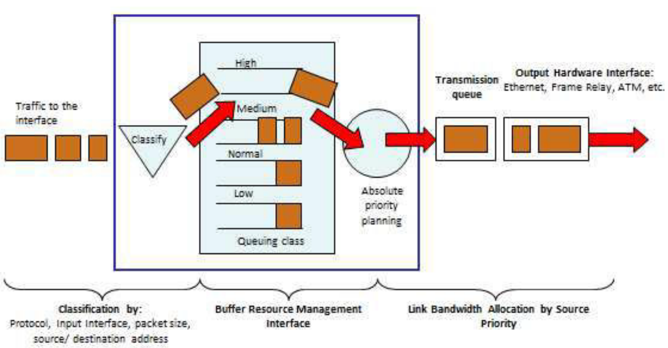


Fig. 1. Initial Priority queuing (PQ) algorithm operation.

C. Custom Queuing (CQ)

CQ queuing has been designed to allow multiple applications or different organizations can share the network with a specific minimum bandwidth or latency requirements. In these environments, bandwidth can be proportionally shared between applications and users. The operation of this algorithm is shown in Figure 2.

There are companies that manufacture devices with characteristics to provide guaranteed bandwidth. These devices can have a remaining bandwidth for additional traffic. CQ queuing handles traffic by allocating a specific number of queue space for each packet class. After this, the use of the queue is based on a round robin strategy.

D. Weighted Random Early Detection (WRED)

Random Early Detection (RED) algorithms are designed to avoid congestion between networks before this becomes a problem. RED algorithms monitor the traffic loads at different points on the network and stochastically drops packets if congestion begins to be increased. Then, the source detects packet loss and reduces its transmission rate. RED algorithm is mainly designed to work with the Transmission Control Protocol (TCP).

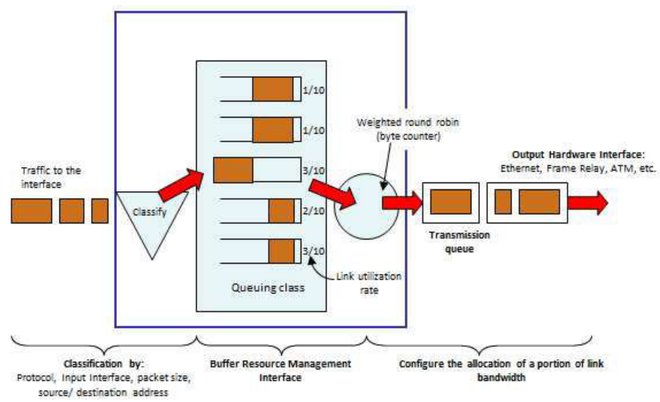


Fig. 2. Custom queuing (CQ) algorithm operation.

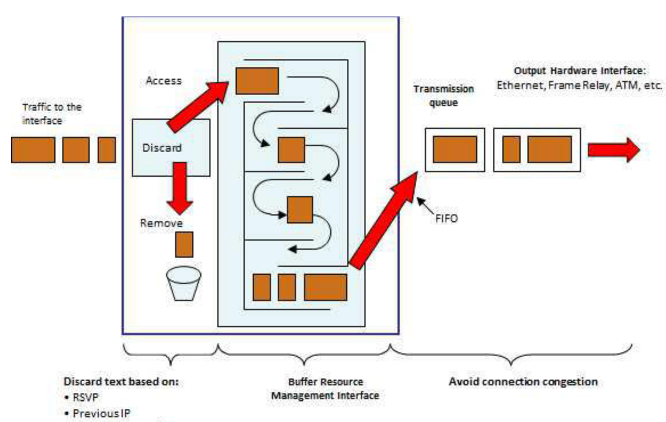


Fig. 3. Weighted Random Early Detection queuing (WRED) algorithm operation.

On the other way, WRED (Weighted Random Early Detection) algorithm combines the capabilities of RED algorithm with IP Precedence or Differentiated Services Code Point (DSCP) or Class of Service (COS) values [4]. This combination makes it easy to handle preferential traffic for high priority packets. WRED can selectively drop lower priority traffic when the interface begins to show signs of congestion. Thus, WRED provides different performance characteristics for different classes of service. This case is shown in Figure 3. The Resource Reservation Protocol (RSVP) using WRED can provide controlled load services.

IV. Operation of the Didactic Tool

Many researchers around the world are integrating information and communication technologies (ICT) in education. In this section we explain the operation of two modules of our digital tool with some examples. First module refers to the FIFO algorithm while the second module refers to the custom queuing algorithm (CQ). Modules for the priority queuing (PQ) and WRED algorithms will be developed in the future to complete our tool. To simulate the FIFO algorithm, a batch with ten packets is placed on the stack, as it is shown in figure 4. First packets that enter are placed at the bottom of the stack, while packets that enter later will occupy the top of the stack. The FIFO algorithm stores packets when the network is congested and forwards them to the network in the order they arrived.

To test the correct operation of the stack we must press FIFO simulation in the simulator. In this way the packets that leave the stack are the first to enter it, while the packets that are going to enter the stack are being formed in the input queue. Figure 5 shows this scenario. In the stack, we can see that only seven packets from original batch remain to be dispatched by the output queue. On the other hand, three new packets in the input queue are waiting to be put on the FIFO stack. The FIFO algorithm also does not provide protection against applications with irregular behavior, so some packets can be invalidated and not stored in the FIFO stack. To evaluate this case, our tool considers a garbage basket where invalid packets (corrupted, lost, etc.) are discarded. This case is shown in figure 6.

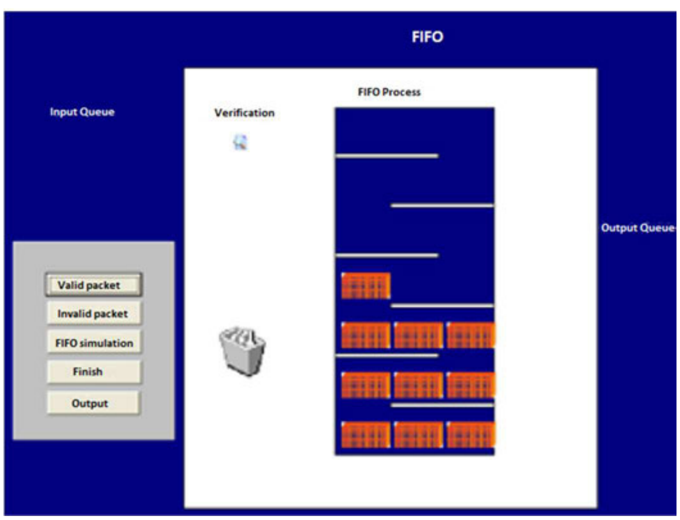


Fig. 4. Ten packets are stored in the FIFO stack.

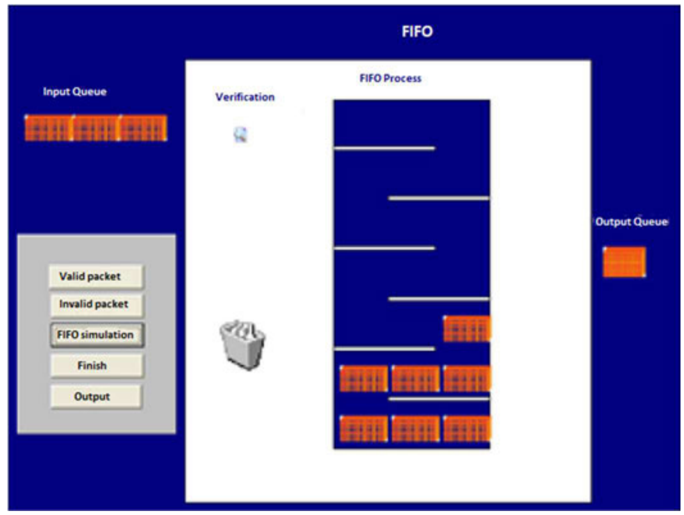


Fig. 5. Evaluating the stack operation.

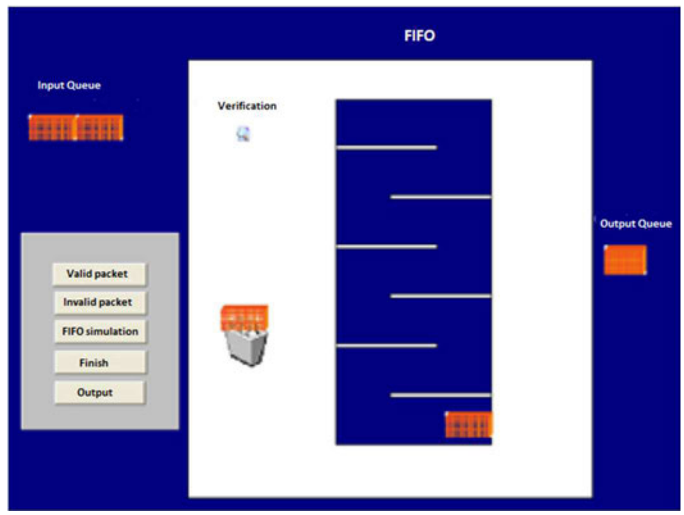


Fig. 6. Evaluating the drop of packets for applications with irregular behavior.

To evaluate the operation of the custom queuing algorithm (CQ), our tool uses six priority queues as we can see in Figure 7. In this case, all queues have equal bandwidth that is equivalent as sending five packets. When a queue sent its five packets, then the next five packets will go to the next queue.

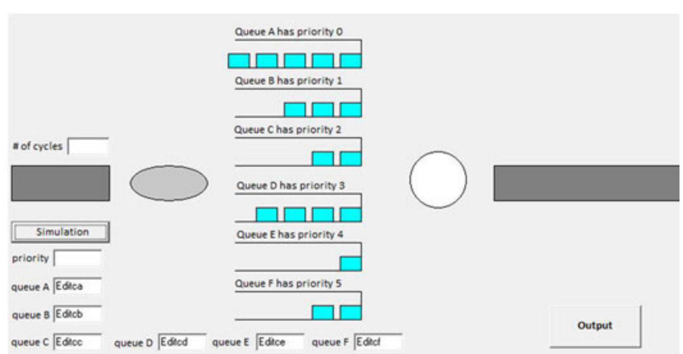


Fig. 7. Scheme to evaluate the custom queuing algorithm.

To start the simulation, we need to define the number of cycles (runtime of simulation). For example, in figure 8 the number of cycles is defined in 12. The simulation is initialized, and packets are allocated to a queue according to their priority. Thus, queues A, B, C, D, E and F have 1, 5, 4, 3, 3 and 1 packets, respectively.

If a queue does not have packets, simulation is moved to the next queue. For outbound traffic, a specific number of queue space is assigned to each packet class and then the queue uses the round-robin strategy. In this case, all the packets of the different queues can be transmitted. Figure 9 shows this scenario. When the number of cycles is 0 the simulation ends.

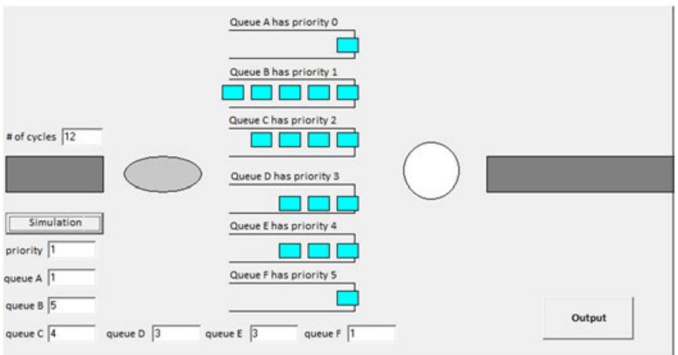


Fig. 8. The number of cycles is fixed in 12 to evaluate the custom queuing algorithm.

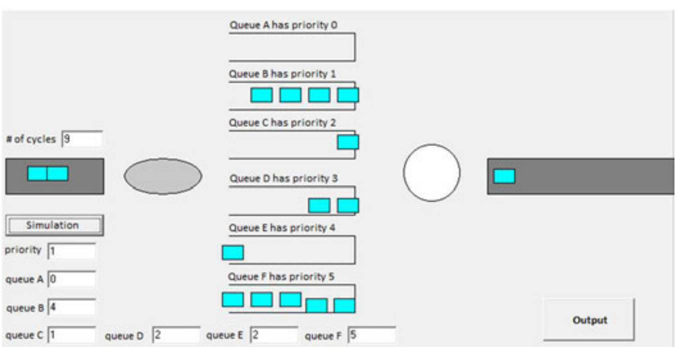


Fig. 9. Operation of the custom queuing algorithm..

V. Conclusions

Information and communication technology have had an important impact on the teaching and learning processes. In computer networking courses, students can find a great deal of material related to concepts and protocols. However, programming and experimentation play an important role to learn about computer networking. This paper tries to highlight the importance of digital tools to support the teaching and learning process about QoS algorithms in the computer networking. Using a digital tool, the students can repeatedly interact with the QoS algorithm operation until the concepts are well assimilated. The animations play an important role to understand the QoS algorithms in a clearer way. Thereby, students can observe how the packets are queued and dispatched using our digital tool. This can represent a support to the teaching/learning based on just reading the algorithms on a printed material. Our experience using this tool has shown us that the students learn algorithms better and faster when interactive material is used. This tool allows the possibility to strengthen the learning about the QoS algorithms both in the classroom as in an autodidactic way. As a didactic resource this tool can be expanded in different directions. As future work new modules for the priority queuing (PQ) and WRED algorithms can be added to this didactic tool.

References

[1] UNESCO, "Enfoque Estratégico sobre las TICs en Educación en América Latina y el Caribe." Santiago, Chile, 2013.

[2] R. Carneiro, J. C. Toscano, T. Díaz, "Los desafíos de las TIC para el cambio educativo," Fundación Santillana, Madrid, 2011.

[3] ITU-T, "Recommendation P.10/G.100 Amendment 3. New definitions for inclusion in recommendation ITU-T P.10/G.100," 2012.

[4] Cisco Systems Inc, "Internetworking Technology Handbook," Fourth edition, Chapter 49, Cisco Press, 2004.

[5] S, Szabolcs, and A. Béla, "A Review of Congestion Management Algorithms on Cisco Routers," Journal of Computer Science and Control Systems, Vol. 5. No. 1, pp. 103-107, 2012.

[6] J.-Y. Le Boudec, and P. Thiran, "Network Calculus, a Theory of Deterministic Queuing Systems for the Internet," Springer, 2001.

[7] H. J. Moya, and D. Roldán, "Comunicaciones en Redes WAN, WiFi, VoIP, Multimedia, Seguridad," Limusa - Noriega Editores, 2006.

[8] R. Raja and P.C. Nagasubramani, "Impact of modern technology in education," Journal of Applied and Advanced Research, pp. 33-35, 2018.

EAI COMPSE 2022

Preliminary Soft-Proceedings

6th EAI International Conference on Computer Science and Engineering

October 6, 2022
Mexico City, Mexico

EAI
RESEARCH MEETS INNOVATION

EAI COMPSE 2022 – 6 October 2022
6th EAI International Conference on Computer Science and Engineering in Health Services

Day 1 – 06/10/2022

9:00 – 9:05 Opening Ceremony
An opening message by the conference organizers.

9:05 – 9:15 Welcome Message by EAI
9:15 – 10:15 Keynote Prof. Utku Kose
Title: Blurred Technology: Roadmaps for Artificial Intelligence and Bioengineering
10:15 – 10:30 Coffee Break
10:30 – 11:50 Session 1: Supply Chain optimization
10:30-10:50 Reverse logistics in recycling companies using a CVRP approach

10:50-11:10 Storage location assignment problem in a warehouse: A literature review

11:10-11:30 A green field analysis for supply chains enhanced with agent-based simulation

11:30-11:50 Resilience in Supply Chains: An Agent-Based Solution Strategy

11:50 – 12:00 Coffee Break
12:00 – 13:00 Session 2: Computational intelligence and computer sciences
12:00-12:20 Intelligent Technology in Geometric Design

12:20-12:40 An Information Architecture for the Engineering and Design of Industrial Electrical Systems

12:40-13:00 Optimized Packing Soft Convex Polygons

13:00 – 14:00 Lunch
14:20 – 15:40 Session 3: Industry 4.0 applications
14:20-14:40 E-Commerce on Startup: A Systematic Literature Review

14:40-15:00 A strategy to analyze the metal packaging market in the food cans industry using Agent-Based Simulation

15:00-15:20 Personalized Emotion detection from text using Machine Learning

15:20-15:40 A State of Art of Non-Fungible Tokens: A Literature Review

15:40 – 15:50 Coffee Break
15:50 – 17:20 Session 4: Health 4.0 and pervasive health

An Information Architecture for the Engineering and Design of Industrial Electrical Systems

Francisco de Asís López-Fuentes[a]

[a] Department of Information Technology, Universidad Autónoma Metropolitana - Cuajimalpa, Av. Vasco de Quiroga 4871, Cuajimalpa, 05348, México City, México.
flopez@cua.uam.mx

Abstract. Electrical systems are one of the most important and necessary facilities that an industrial plant has. An industrial electrical system consists of various components such as conductors or cables, connectors, switches, electrical circuits, etc. During the design of an industrial electrical system, engineers and designers require a great deal of technical information. It is a tedious process for an electrical engineer because of the need to collect information from different digital or printed catalogues. In this situation, an information system with information all components of an electrical system can help to designers and engineers in their tasks. This paper presents an information architecture to build this information system. Our proposed architecture will allow to acquire, analyze and monitor data in the way that the user requires.

Keywords: power systems, information systems, data modeling, electrical design.

1 Introduction

Today the industrial companies face many challenges to diversity their products, increase the resource efficiency and to improve time to market [1]. The fourth industrial revolution has defined a new level of organization and control in the life cycle of products [2], [4] which affects different areas in a company. The Industry 4.0 concept presents different challenges in the industry and the companies should review their designing and drafting methods in all their disciplines to base the development of their products in a modern and interdisciplinary approach [3]. The growth of the digitization, IT penetration, and networking have increased these challenges in the industry. Digitization of the designing and engineering processes for industrial plants have an important role in the today's industry. Industrial or residential infrastructure projects generally involve a variety of facilities. Within these infrastructures, electrical installations have an important role. The design of an electrical system involves modeling and design calculations of lighting, power and cable routing systems, panel board design and balancing, lighting protection, cost estimation, installation detailing, layout planning. During the design processes, engineers require technical information on the various components as well as their characteristics. During the design of an electrical distribution system for a building, industrial plant or residential area is required to observe norms and standards of the electrical industry such as NEC, ANSI, NEMA [2], [3]. This activity is useful to define the concepts, components, electrical symbology, calculation recommendations and electrical design. Currently, there are methodologies as BIM (Building Information Modelling) for the digitization of the design and engineering processes of the industrial plants [7]. However, government agencies and researchers need to do significant efforts for the standardization of BIM methods related with projects of electrical engineering. Also, object modeling in BIM must be compatible with manufacturer-based analysis and design studies [8]. In this paper, we present an information architecture for planning and design electrical industrial systems through the selection of the most important electrical components. Our proposed architecture tries to provide a solution to develop specialized CAD/CAE systems and more focused on small groups of design engineers in the planning of electrical installations

The rest of this work has the following organization. Section 2 gives an overview about the different stages for the planning and design of an industrial electrical system. Conceptual architecture design with the interaction between the different programs and the database are presented in Section 3. Interaction between the different types of programs and the files are presented in Section 4. The data modelling and the relationships between entities based on the entity-relationship model are discussed in section 5. Our article concludes in Section 6.

2 Planning and design of an industrial electrical system

To define the information architecture and the data modelling of an industrial electrical system we need to define the electrical design process for an industrial plant. The most common planning and development for an electrical project has the following stages [5], [6]:

1. Development of a preliminary project with the distribution of equipment, machinery and the production areas in the industrial plant.
2. Definition of the electrical load zones within the industrial plant based on the distribution of the equipment to be connected.
3. Planning of the design of the electrical distribution system based on the following procedure: examination of the load, analysis of the demand in its different divisions (demand, peak load, demand factor, maximum demand, diversity factor, demand factor, load factor, coinciden demand), investigation and selection of the appropriate distribution system for the plant and its representation in a single line diagram with the project loads.
4. Definition of the type of distribution system to be used (simple radial, expanded radial, primary system, secondary system, ring bus, etc.)
5. Definition of the following aspects:
    a) distance from load centers to the equipments
    b) characteristics of the power lines
    c) loads of the main feeders and branch circuits

d) measurement and protection equipment according to the characteristics of the load.
e) capacities of the electrical equipment and elements involved in the system.

6. Once the capacity of the electrical elements involved in the project has been defined, their physical dimensions are defined and the physical location is carried out within the distribution plan of the industrial plant equipment.
7. Once the distribution of electrical equipment and feeders is completed, a catalog of specifications is made with the construction characteristics of each element involved.

In the planning and design of an electrical system, it is necessary to define the elements that constitute it.

3 Conceptual architecture design

The general conceptual criteria for this architecture is as follows:

As a first step, we propose to create a database with relations (tables) based on entity-relationship diagrams. These tables are used by the queries required from the different application in the information system. The entities and relationships tables between entities contain the attributes of the electrical elements for an industrial electrical system, and will allow us to make direct queries, such as the following:

- Characteristics of a conductor (e.g.: weight, price, resistance, etc.).
- Characteristics of a transformer (e.g: manufacturer, capacity, price, etc.).

In the database we can make direct queries between entities, relating the different entities, these queries could be as follows:

- Select an electrical switch for a transformer based on its voltage.
- Consult different types of electrical switches for a transformer.
- Consult the different electrical protections for a conductor.

In the second step, we can develop different applications to access the database. These applications are programs which perform calculations to determine the capacities of the conductors for an industrial electrical system. The different data required by the application programs such as resistance, area, capacity of the conductor are extracted from the database. Once the calculation process is finished in the application programs, we obtain optimal values with which we can independently access the database and select the appropriate cable. This process is illustrated in figure 1.

For the handling and design of the graphic parts of an equipment or a component such as side views, front views, dimensions, photographs, two options will be considered (see figure 2):

1. To use an own CAD, which allows to draw different schemes or equipment components. Subsequently, these drawings can be saved in a graphic format such as GIF (Graphics Interchange Format) or another.
2. Import external and independent files from an external CAD to the system. This will allows that drawings done by an external CAD system can be managed by the information systems (e.g. image file formats or graphic formats). Therefore, these drawings or images can be invoked by application and interface programs and can be displayed where they are required. This facilitates the construction of graphic parts from different CAD systems based on industry standards.

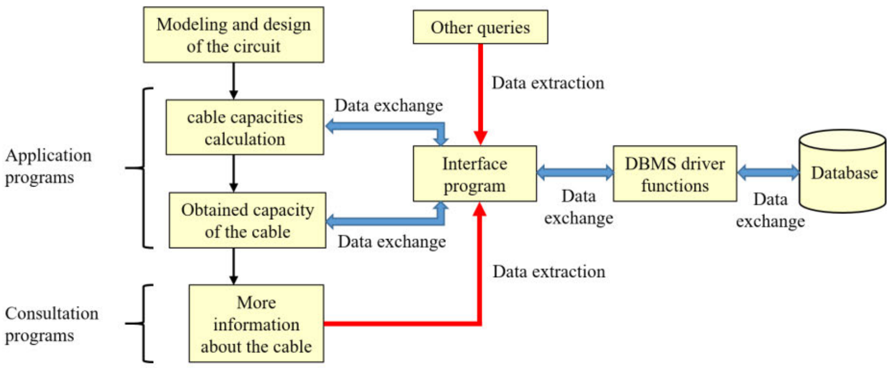


Fig.1. Interaction between application programs and databases.

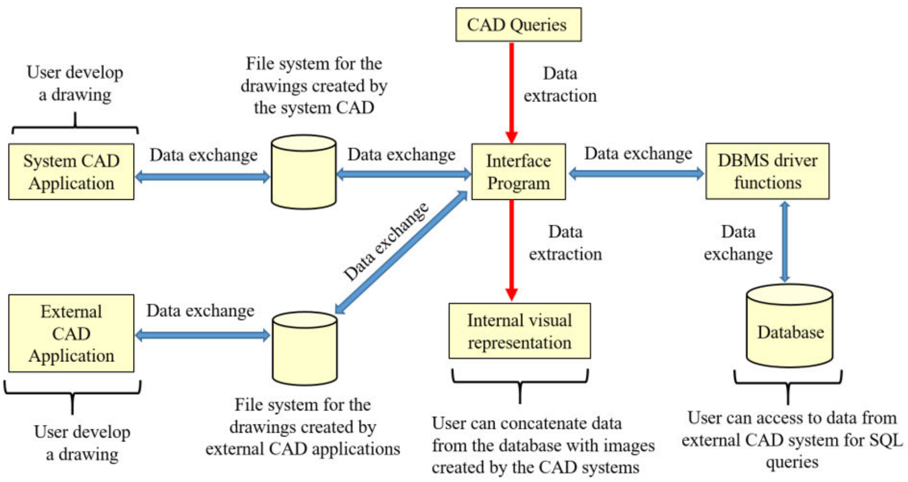


Fig.2. Interaction between CAD applications, database and the graphical interface.

When a query is done to know the graphical characteristics of an equipment or device, the interface program invokes a function that allows visualizing the graphic characteristics of the equipment with data extracted from the database. In this way, we can obtain a technical sheet with the specifications of the device or equipment. Also, the system will be able to query the different units, without necessarily having to invoke their graphic features. With the CAD system will be possible to build drawings for electrical systems using symbols from the CAD system itself, and save these drawing in some predefined format (e.g. GIF). This allows the manipulation of these drawings in other information systems that support this type of format. Database tables can be accessed from an external CAD system to perform SQL queries or applications with different drawing formats.

4 Interface with the programs

For our information architecture, we consider three different types of programs, which have been defined based on their functionality:

- *Application programs:* these programs allow to obtain a specific result for a particular use. In this case, the application programs will be for the electrical calculation algorithms of conductors. The results obtained from these programs allow an adequate selection of the electrical conductor using three different methods.
- *Query programs:* These are programs to access the database. Unlike application programs, query programs will not require returning any data to be processed by any program. The requested data only is displayed on the screen for the user. These programs are also responsible for invoking the graphical characteristics of the entities (equipment and electrical components) from the database.
- *Interface programs:* These are the programs used to open and access the database. In our case, these programs will be used each time data is required, either by the application programs or by the query programs.

The interaction of these programs with the database is divided into different steps which allow controlling access to the database and its tables through the DBMS (Database Management Systems). These steps are as follows:

1. *Connection to databases.* This is the way to access a database. The connections to the database are required before any queries can be performed.
2. *Opening the communication.* This action is performed by the DBMS driver of the corresponding database. This step opens the communication and the table to be consulted in the database.
3. *Compilation of the original declaration.* In this step, a character data is obtained from the database that will be used in a query program.
4. *Data extraction or reading*. It is the next step after the request is executed. The purpose of this step is to move tuple by tuple through the resulting set of tuples, select the current tuple, and place it in the output buffer.
5. *Closing the communication.* This is done by the DBMS driver to close the communication and the queried database table.

Figure 3 shows the interaction between these different programs and the different types of files in our information architecture.

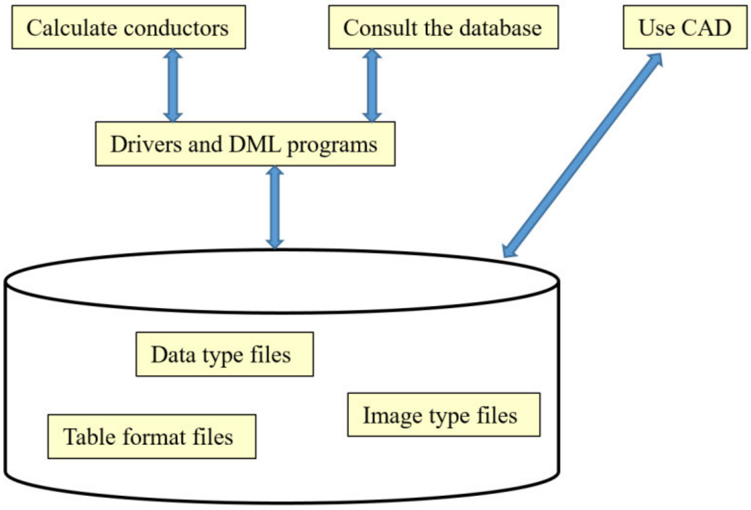


Fig.3. Interaction between the different programs and the different types of files.

In this architecture, there are three application programs. The first performs the selection of conductors. The second program allows data to be extracted from table-type files and associated with the corresponding image files, while the third program performs the CAD operations. The selection of electrical conductors is made based on the procedures of current capacity, voltage drop and short circuit, and according to the NEC regulations. The calculation of conductors in an electrical system is important because its correct selection allows the adequate flow of electrical energy from a primary source to the final load. In this way, the conductors distribute energy to the different electrical loads in an industrial plant. If a user requires a conductor analysis, then the *"Calculate conductors"* option is chosen. However, we can see in figure 3 that the database is used by the "calculate conductors" program, as well as the query programs (*Consult the database*). Depending on the needs of the user, the corresponding option will be chosen, and the files will be required in a different way. For example, the "calculate conductors" option requires the use of DML (Data Manipulation Language) programs and drivers because this program will interact with the database tables, extracting data from the table format files in order to calculate different operations. If the user decides to make diagrams or drawings of the electrical system, then the option "Use CAD" must be selected. In this architecture, using the CAD option, the user can make diagrams and equipment drawings, as well as save the images made in a defined image format. The "Consult the database" program queries the different elements of an electrical system and allows knowing the technical data, images, diagrams, dimensions of the corresponding equipment. For example, if a user searches for the motor element of a specific capacity, the program accesses the database, extracts the motor information and associates it with the corresponding image.

5 Data modeling

For the data modeling, the following technical points have been considered:

- Design the conceptual part of the database, such as the following: define the entities, the attributes of each entity, the relationships, the ER diagram (Entity - Relationship) [9], [11], the different affinities based on the relational model, the normalization required for these affinities and the data dictionary.
- Also for the design criteria of the database, it is necessary to consider the relationships that the devices and components of an industrial electrical system have among themselves, the recommendations of manuals in the electrical industry, the recommendations of the equipment manufacturers, and the diagrams. In this data modeling was considered the following standards: NEC, NEMA, IEEE, ANSI [10].

In the data modeling of this work, the relationships are the associations that can be established between different entities [9], [12]. In this work the main components of an industrial electrical system are represented as entities. An E-R diagram used to build the information architecture is shown in figure 4. Here, all entities that are involved in the design of the electrical system are considered. In this diagram, the attributes of the entities and the relationships are not indicated.

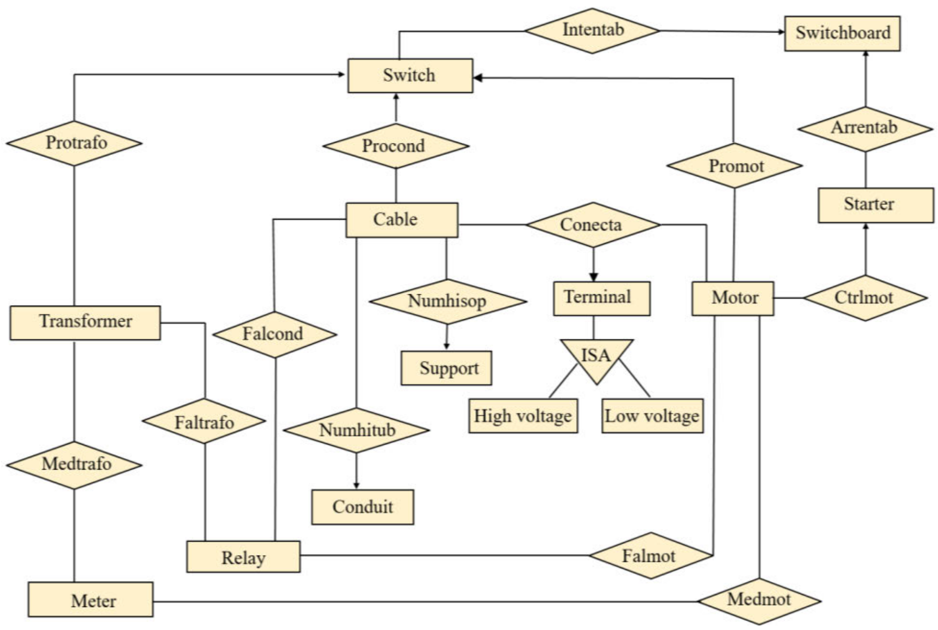


Fig.4. Diagram E-R used as reference to build the information architecture

6 Conclusions

An information architecture plays an important role for the digitization of the design and engineering processes of industrial electrical systems. In this paper, we present an information architecture for build a information system which helps to the engineers to select the equipments during the electrical design processes of an industrial plant. Our proposed architecture integrates information generated from three different aplications program with a database. These application programs are used to conductors calculation, computer aided design, and database consultation. Our conceptual design shows the interaction between the application programs and databases. Also, there are interaction between different programs and different types of files. In our architecture, the data are modelling using the E-R model, where each principal electrical component of an industrial electrical system is modelled as an entity. We believe that our current work can be extended to other fields related to industry 4.0. We plan to do this activity as future work.

References

1. Rennung, R., Luminosu, C. T. & Draghici, A.: Service Provision in the Framework of Industry 4.0, SIM 2015 / 13th International Symposium in Management, Procedia - Social and Behavioural Sciences 221, 372 – 377 (2016).
2. Vaidya, S., Ambad, P., & Bhosle, S.: Industry 4.0—Glimpse. Procedia Manufacturing, 20, 233–238 (2018).
3. Eigner, M.: Modellbasierte Virtuelle Produktentwicklung auf einer Plattform für System Lifecycle Management, in Industrie 4.0- Beherrschung der industriellen Komplexität mit SysLM, Berlin, Heidelberg, Springer (2013).
4. Oztemel, E., Gursev, S.: Literature review of industry 4.0 and related technologies. J. Intell. Manuf. (2018).
5. IEEE: IEEE Recommended Practice for Electric Power Distribution for Industrial Plants, ANSI/IEEE Std 141-1993 (1993).
6. Short, T. A.: Electric Power Distribution Handbook, Boca Raton, FL, CRC Press (2004).
7. Eastman, C., Tiecholz, P., Sacks, R., Liston, K.: BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors (2nd ed.). Hoboken, New Jersey, John Wiley (2011).
8. Farooq, J., Sharma, P. and Kumar, S.: Applications of Building Information Modeling in Electrical Systems Design. Journal of Engineering Science & Technology Review, 10(6) (2017).
9. Silberschatz, A, Korth, H, Sudarshan, S.: Database systems concepts, seventh edition, McGraw-Hill, USA (2019).
10. Jay, F.: IEEE Standard Dictionary of Electrical and Electronics Term, third edition, IEEE Press, NY, USA (1984).
11. Codd, E. F.: The Relational Model for Database Management: Version *2*, first edition, USA: Addison-Wesley Pub. Co, 525p, (1990).
12. Kroenke, D.; Database Processing, Fundamentals, Design and Implementation, 5th edition, México: Prentice-Hall Inc. 605p, (1995).