

Informe final

Proyecto de grado Logística Humanitaria

Integrantes

Martín Fernández Staehle

Laura Aizpún Fernández

Tutores

Omar Viera

Sandro Moscatelli

Departamento de Investigación Operativa

Instituto de Computación

Facultad de Ingeniería

Universidad de la República Oriental del Uruguay

2013



RESUMEN

Se vincula a la logística humanitaria, con las actividades logísticas que tienen lugar en el marco específico de la acción humanitaria de emergencia. Principalmente se tiene como objetivo atender las necesidades de las personas afectadas por una situación de emergencia o desastre y minimizar su sufrimiento. Para alcanzar este objetivo, se pueden aplicar conceptos de logística humanitaria para que las acciones y las operaciones sean más rápidas, ágiles y efectivas.

Este proyecto presenta una investigación sobre el tema que incluye: aspectos que abarca la logística humanitaria, modelos y algoritmos asociados a la temática, diferencias que existen entre logística humanitaria y logística comercial y un prototipo de software que intenta aplicar algunos los conceptos adquiridos.

Para poder realizar estas actividades, se realiza un relevamiento de modelos y algoritmos, a partir de los cuales, se adapta un modelo matemático y se construye una heurística para la implementación del mismo.

Palabras clave: Logística Humanitaria, Gestión de Riesgos, Distribución de suministros, Ayuda Humanitaria.

CONTENIDO

Resumen.....	2
1 Introducción	7
1.1 Contexto	7
1.2 Motivación.....	7
1.3 Objetivos del proyecto	8
1.4 Resumen del problema.....	8
2 Marco Teórico	11
2.1 Introducción.....	11
2.2 Gestión de riesgos.....	11
2.2.1 Etapas gestión de riesgos	12
2.3 Logística humanitaria	13
2.3.1 Gestión de inventarios.....	14
2.3.2 Distribución de suministros y recursos	15
2.3.3 Evacuaciones	15
2.3.4 Albergues temporales	16
2.4 Software para la Logística Humanitaria	16
2.4.1 LSS.....	17
2.4.2 Sahana	18
2.4.3 HELIOS.....	19
2.4.4 HFOSS.....	19
2.4.5 DMIS	20
3 Definición del Problema	21
3.1 Introducción.....	21

3.2	<i>Antecedentes</i>	21
3.3	<i>Hipótesis o supuestos</i>	22
3.4	<i>Respuesta rápida</i>	23
3.5	<i>Problema a resolver</i>	23
4	Descripción de la solución	26
4.1	<i>Alcance</i>	26
4.2	<i>Requerimientos</i>	27
4.2.1	Funcionales	28
4.2.2	No Funcionales	29
4.3	<i>Modelo de distribución</i>	29
4.4	<i>Diseño y funcionamiento de la heurística</i>	32
4.5	<i>Herramientas utilizadas</i>	38
4.5.1	Java	38
4.5.1.1	JDK	38
4.5.1.2	Netbeans	38
4.5.1.3	Java EE	39
4.5.1.4	GlassFish Server	39
4.5.1.5	JDBC	39
4.5.1.6	EJB	39
4.5.1.7	JPA	39
4.5.1.8	JSP	40
4.5.1.9	jQuery	40
4.5.1.10	JSON	40
4.5.2	MySQL	40
4.5.3	Google Maps API	41
4.5.4	GLPK	43
4.5.5	Assembla Workspace	43
4.6	<i>Componentes de la solución</i>	44

4.6.1	Arquitectura del sistema	44
4.6.1.1	Capa cliente	45
4.6.1.2	Capa web.....	45
4.6.1.3	Capa de negocio	45
4.6.1.4	Capa de datos	46
4.6.2	Estructura y funcionamiento de la implementación.....	46
4.6.2.1	ProyectoLH-ejb	46
4.6.2.2	PriyectoLH-war	47
4.6.3	Modelo de la Base de datos.....	48
4.6.3.1	Mapa.....	49
4.6.3.2	Nodo	50
4.6.3.3	RelacionNodo	50
4.6.3.4	Distribución	50
4.7	<i>Proceso de desarrollo</i>	51
4.7.1	Análisis del problema	51
4.7.2	Investigación	51
4.7.3	Diseño de la solución.....	52
4.7.4	Implementación	52
4.7.5	Integración	52
4.7.6	Pruebas	53
4.7.7	Identificación y planificación de tareas para el desarrollo.....	53
5	Plan de pruebas	55
5.1	<i>Metodología</i>	55
5.2	<i>Pruebas corridas y resultados obtenidos</i>	56
5.2.1	Pruebas del modelo utilizando solver GLPK	56
5.2.2	Pruebas Heurística.....	59
5.2.3	Pruebas del Sistema	62
6	Conclusiones	65

7	Trabajo Futuro y posibles mejoras.....	67
8	Anexos	70
8.1	<i>Modelo de Distribución (suministros individuales)</i>	<i>70</i>
9	Referencias	75

1 INTRODUCCIÓN

1.1 CONTEXTO

Los desastres naturales o provocados por el hombre, son eventos de baja probabilidad de ocurrencia pero provocan alto impacto. La Gestión Integral de Desastres (Riesgos) es una disciplina que se encarga de la gestión de desastres. En particular, en la fase de Respuesta y en la de Recuperación las actividades más importantes y las más costosas (hasta un 80 % de los costos) están relacionadas con la denominada Logística Humanitaria.

La gestión de emergencias se caracteriza por estar asociada a recursos limitados, falta de organización, y es allí donde entra la logística que se encarga de optimizar el uso de los recursos, e intenta que las acciones y operaciones sean más rápidas, ágiles y efectivas.

Los desastres repentinos que requieren una rápida movilización de recursos materiales y humanos para atender a las víctimas de la catástrofe. Se vincula a la Logística Humanitaria, las actividades logísticas que tienen que ver con las acciones humanitarias de emergencia, y tiene como primer objetivo, minimizar el sufrimiento de la población afectada.

1.2 MOTIVACIÓN

Cada año, desastres naturales afectan a miles de personas en Uruguay y el mundo. Generalmente, el mayor impacto asociado a estos desastres lo sufren personas que viven en zonas vulnerables, quienes a su vez no están familiarizados con los procedimientos o métodos a seguir en estas situaciones, todo esto conlleva a que las acciones humanitarias de ayuda sean necesarias y además, lo más conveniente es que sean efectivas y eficientes.

En Uruguay, el Sistema Nacional de Emergencias (SINAE) [31] es el encargado de atender situaciones de emergencias, crisis y desastres de carácter excepcionales, que afecten o puedan afectar en forma significativa al Estado, sus habitantes o los bienes de los mismos y que excedan la posibilidad de ser resueltos por los organismos u órganos operando en el ámbito normal de sus competencias.

Un fenómeno frecuente que se da en Uruguay, son las inundaciones [63]. Existen distintos tipos de inundaciones, algunas ocurren cuando por ejemplo un río se desborda, y otras que son más repentinas, las mismas pueden desarrollarse rápidamente debido por ejemplo a lluvias torrenciales. Las grandes lluvias son la causa principal de inundaciones, pero además hay otros factores importantes que provocan que los efectos de las inundaciones se vean agravados, como algunas actividades humanas, sobre todo a causa de la contaminación del medio ambiente.

De esta manera, resulta necesario relevar estudios realizados y aplicados por parte de distintas organizaciones alrededor de todo el mundo en torno a la temática para poder mejorar la acción humanitaria en este país.

1.3 OBJETIVOS DEL PROYECTO

En esta sección se marcan los siguientes objetivos puntuales:

- Realizar un relevamiento de las definiciones, modelos, metodologías y herramientas existentes relacionadas con la Logística Humanitaria.
- Propuesta de modelos y algoritmos posibles de ser aplicados en Uruguay.
- La implementación de un prototipo que permita planificar, controlar y modificar en tiempo real la planificación logística.

1.4 RESUMEN DEL PROBLEMA

Se plantea realizar una investigación con el objetivo de hacer un estado del arte sobre Logística Humanitaria. Se busca relevar los modelos existentes desarrollados en los últimos años y analizar las principales cuestiones que se abarcan dentro de la temática seleccionada. Esto incluye el análisis de algoritmos y discusión de alternativas.

Comprender las diferencias entre logística humanitaria y logística comercial, para poder reutilizar y aplicar la amplia gama de metodologías y herramientas desarrolladas ya existentes relacionadas a la logística comercial.

En el contexto presentado, se plantea además, resolver un problema de decisión operativa a través de una nueva herramienta de software. La idea principal que inspira los requerimientos del prototipo de desarrollado, es la de acelerar la toma de decisiones

a corto plazo, por parte de los responsables en atender emergencias, con el objetivo principal de generar una respuesta en el menor tiempo posible. En definitiva, se propone implementar un sistema de información que permite agilizar el proceso de toma de decisiones sobre la forma de distribución de suministros en la etapa de respuesta a una emergencia.

Tomando como base el modelo de distribución planteado en el documento Estado del Arte (punto 8.1), se construyó un nuevo modelo que se adapta mejor a la realidad en la cual se lo quiere aplicar. Se eliminan y modifican algunas restricciones así como también el objetivo de optimización para buscar minimizar el tiempo de la acción humanitaria de respuesta. Además, a este modelo se le agregó la restricción de que se debe satisfacer toda la demanda existente, mas delante se describe el comportamiento específico relacionado con la restricción de demanda. A su vez, también se implementa un algoritmo (heurística) que resuelve el problema que ataca el modelo mencionado anteriormente; el software desarrollado utiliza principalmente esta heurística para hallar las soluciones.

Este sistema permite resolver el problema de distribución de suministros y o personas desde ciertos puntos de partida hacia donde sean necesitados, buscando minimizar los tiempos necesarios para llevarla a cabo. Determinando cuales destinos se visitan desde que orígenes, en qué orden y cuantas unidades del suministro es transportado en cada tramo y por consecuencia, cuántas unidades se dejan en cada sitio.

Bajo el supuesto de que se conoce los lugares geográficos donde se encuentran las personas afectadas por un desastre, un estimado de la cantidad de kits de suministros existentes que se necesitan distribuir y los lugares geográficos donde conseguirlos. Se define un conjunto de proveedores, un conjunto de demandantes que representan el lugar geográfico en el que hay personas que necesitan suministros. Estos suministros pueden llegar a ser de cualquier índole, como por ejemplo un set con alimentos, agua y ropa. A priori, los conjuntos podrían ser muy grandes y puede llegar a ser muy complicado decidir cómo y en qué orden visitar a las zonas necesitadas. El sistema pretende resolver en poco tiempo de ejecución, la forma de distribuir todos los suministros y garantizar que la distribución sea aproximadamente óptima en términos del tiempo transcurrido desde que los vehículos salen del origen hasta que llegan a la zona afectada.

En una primer etapa, se realizó un prototipo para hacer pruebas con el solver GLPK [18] para comprobar que realmente era posible encontrar soluciones en tiempos de procesamiento aceptables. Al encontrar que para ciertos casos particulares de entradas con gran cantidad de nodos, no se obtuvieron buenos resultados, se desarrolla un algoritmo (una heurística) que intenta resolver el problema de distribución en un tiempo computacional breve.

Al momento de calcular una solución, en primer instancia, se intenta resolver el problema con el solver de programación lineal GLPK [18]. Si este proceso no devuelve una solución en N segundos (valor configurable), se procede a la ejecución de la heurística.

Dados los datos de una situación de emergencia, puede ocurrir que existan casos particulares en los cuales la cantidad de suministros disponibles no alcance para satisfacer toda la demanda. Para estos casos, no va a existir una solución factible tomando en cuenta las restricciones del modelo y el algoritmo (heurística) encuentra una distribución aproximadamente óptima y equitativa.

2 MARCO TEÓRICO

2.1 INTRODUCCIÓN

En esta sección se plantea un resumen de algunos de los puntos tratados en el documento Estado del Arte, abarcándose los temas de Gestión de Riesgos, Logística Humanitaria, Gestión de Inventarios, Evacuaciones y Albergues Temporales.

A su vez, se mencionan algunas de las principales herramientas existentes en la actualidad que se utilizan para facilitar y llevar a cabo de manera más eficiente diversas tareas asociadas a la logística humanitaria.

2.2 GESTIÓN DE RIESGOS

La Gestión de Riesgos es la disciplina que se encarga de la gestión de desastres, teniendo como objetivo el planeamiento, administración y control de actividades de todo lo referente al accionar durante y después de ocurrido un desastre. De esta manera, la Gestión de Riesgos busca reducir el impacto del desastre en la zona afectada (enfocándose principalmente en el aspecto humanitario), así como también disminuir la probabilidad de ocurrencia del mismo [30].

En Uruguay, la entidad encargada de coordinar los recursos y acciones a llevarse a cabo ante la ocurrencia de un desastre es el SINAE (Sistema Nacional de Emergencias); el mismo está conformado por un conjunto de organismos (tales como ser el Poder Ejecutivo y la Dirección Nacional de Emergencias, entre otros), y su objetivo según la ley N°18621 [31] es “la protección de las personas, los bienes de significación y el medio ambiente, ante el acontecimiento eventual o real de situaciones de desastre (...) de modo de propiciar las condiciones para el desarrollo nacional sostenible” [31].

Existen varios modelos que abordan la Gestión de Riesgos, uno de los más conocidos y utilizados es el modelo tradicional, el cual se basa en un proceso secuencial conformado por un conjunto de fases y actividades a ejecutarse tanto antes como después de ocurrido un desastre [32].

Otro modelo para la Gestión de Riesgos que es importante mencionar es el de expansión-contracción, el cual aborda la gestión de riesgos como un proceso continuo, donde las actividades correspondientes a las distintas fases no cesa en ningún momento, aunque el accionar de cada una de ellas varia (intensificándose o disminuyendo) en función del tiempo y de la situación en la que se encuentre la comunidad afectada [33]. El modelo tradicional divide la Gestión de Riesgos en dos fases, antes de que ocurra el desastre y después de ocurrido el mismo.

Las etapas a desarrollarse en la fase anterior a la ocurrencia del desastre son [34]:

- Prevención
- Mitigación
- Preparación
- Alerta

Mientras que las etapas que se llevan a cabo una vez ocurrido el desastre son [34]:

- Respuesta
- Rehabilitación
- Reconstrucción

El presente trabajo se enfoca principalmente en el modelo tradicional debido a que es uno de los más utilizados, abarcando cada una de las etapas asociadas a la gestión de riesgos; a su vez, para un mejor entendimiento de la gestión de riesgos, permite identificar y separar claramente las acciones y acontecimientos que están asociados a cada una de las etapas del mismo.

2.2.1 ETAPAS GESTIÓN DE RIESGOS

La etapa de Prevención tiene por objetivo determinar las posibles amenazas y vulnerabilidades que pueden afectar la zona, detectando vulnerabilidades presentes en la comunidad que puedan poner en riesgo a las personas que allí residen en caso de que se produzca el desastre. De esta manera, es necesario realizar estudios de la zona y recabar datos históricos que permitan identificar dichas amenazas y vulnerabilidades, para luego poder actuar en consecuencia [32].

El propósito de la Mitigación es reducir las vulnerabilidades existentes en una comunidad, de manera de modificar las características de la amenaza, logrando disminuir en la mayor medida posible el impacto de la misma sobre la comunidad [35].

La etapa de Preparación, busca organizar y estructurar la respuesta ante la ocurrencia de un desastre, de manera de lograr actuar lo más rápidamente posible; para tal fin, es necesario planificar de antemano acciones a tomarse referentes a temas tales como búsqueda, salvamento y asistencia de afectados, así como también elaborar inventarios de recursos, entre otros temas. [35]

En la etapa de Alerta se especifican los estados de alerta, los mismos sirven para prevenir a la comunidad sobre la inminente ocurrencia de un determinado desastre, de manera de poder tomar determinadas acciones en pro de reducir potenciales daños. La posibilidad de alertar a la población tiene relación directa con el tipo de desastre y la probabilidad de predecir la ocurrencia del mismo [35].

La etapa de Respuesta entra en acción inmediatamente después de ocurrido el desastre, llevándose a cabo actividades como la búsqueda, rescate y evacuación de afectados, se levantan o habilitan los albergues temporales, y se realiza la distribución de suministros correspondiente. El principal objetivo aquí es actuar lo más rápido posible buscando disminuir el sufrimiento de los afectados [36].

La Rehabilitación consiste en la restitución, lo más rápido que sea posible, de los servicios básicos e indispensables para el correcto funcionamiento de la comunidad, como ser los servicios de salud, energía, agua potable y caminos, entre otros [30].

Finalmente, en la etapa de Reconstrucción se busca llevar a la comunidad a un estado de desarrollo igual o superior al que se encontraba al momento de ocurrido el desastre; para esto es necesario llevar a cabo procesos de mediano y largo plazo, buscando reflotar la economía (trabajo e industrias), reparar la infraestructura dañada (viviendas, caminos), brindar ayuda social (apoyo psicológico a las familias afectadas), etc. [30] En esta etapa es importante implementar las medidas provenientes de los planes de prevención y mitigación a la vez de aplicar las enseñanzas o conclusiones obtenidas tras la ocurrencia del desastre, buscando reducir las vulnerabilidades de la región y así disminuir el impacto tras el suceso de un nuevo desastre [30].

2.3 LOGÍSTICA HUMANITARIA

La logística humanitaria entra en acción en la etapa de respuesta, abarcando disciplinas tales como la distribución de suministros y recursos, gestión de inventarios y almacenamiento de suministros, así como también la localización y manejo de albergues temporales, y planificación de evacuaciones.

Es necesario realizar una planificación de las tareas y acciones a tomarse, así como también definir responsables y recursos para cada una de las disciplinas mencionadas anteriormente, de manera de brindar acciones de respuesta no solo rápidas, sino también ágiles y efectivas, de manera de minimizar al máximo el sufrimiento de las víctimas.

La diferencia principal entre la logística comercial y la humanitaria, desde el punto de vista de la distribución de suministros, puede encontrarse en los fines que ambas

persiguen; mientras la logística humanitaria busca atender la población afectada (buscando minimizar su sufrimiento), la logística comercial se enfoca en maximizar beneficios (generalmente económicos) [37].

Por otra parte, la logística humanitaria se caracteriza por buscar soluciones rápidas (se busca actuar lo antes posible) y que logren llevar la ayuda lo antes posible a la zona afectada (minimizando el tiempo) sin considerar el factor económico como uno de los objetivos principales a minimizar, mientras que en la logística comercial uno de los intereses principales es el de minimizar costos, primando la maximización de ganancia por sobre la rapidez de una solución [37].

2.3.1 GESTIÓN DE INVENTARIOS

La gestión de inventarios, corresponde a llevar un control y seguimiento de las reservas existentes, para ser utilizados cuando correspondan en algún punto de la cadena de suministros, siendo un factor clave en la planificación y operación de los sistemas logísticos en general.

El fin de la gestión de inventarios, en general, es el de reducir costos logísticos (no solo monetarios), ayudando a satisfacer las demandas en tiempo y forma, brindando por ejemplo la posibilidad de saber cuánto mantener en inventarios así como también cuándo realizar reposiciones del mismo [54].

En la Logística Humanitaria, a diferencia de la logística comercial, la gestión de inventarios debe resolver el problema de que la demanda no siempre es constante ni se mantiene dentro de determinados márgenes, todo lo contrario, se tiene una demanda completamente aleatoria y extremadamente variable de un momento a otro (por ejemplo ante la ocurrencia de un desastre la demanda de determinados productos básicos se dispara) [49].

En consecuencia, resulta muy difícil saber cuánta cantidad de suministros mantener en inventario (depende de la magnitud del desastre), así como también qué suministros tener (diferentes tipos de desastres pueden requerir distintos suministros) y cuándo reponer (nunca se sabe cuándo ocurrirá el próximo desastre; posibles suministros perecederos).

Existen múltiples modelos que abordan y brindan soluciones que ayudan a manejar la gestión de inventarios en general, uno de los más interesantes es el que plantean Ghiani et al. [39], el cual se describe y desarrolla en el documento Estado del Arte [Estado del Arte - punto 3.4]. Existen modelos más específicos desarrollados para el contexto de logística humanitaria que resultan ser más complejos porque contemplan aspectos

propios y particularidades de las situaciones de emergencia.

2.3.2 DISTRIBUCIÓN DE SUMINISTROS Y RECURSOS

La distribución de suministros y recursos en el contexto de la Logística Humanitaria busca realizar una distribución lo más rápida posible, de manera que los suministros (alimentos, vestimenta, etc.) y recursos (ayudantes, médicos, etc.) puedan llegar a las zonas afectadas en el menor tiempo posible. Por lo tanto, los objetivos principales son diseñar y ejecutar una distribución que satisfaga en la mayor medida posible la demanda de las zonas afectadas y a su vez que logre satisfacerla en el menor tiempo posible [40].

Uno de los problemas más importantes a considerar al momento de planificar y diseñar la distribución en una región donde ha ocurrido un desastre es que muchas veces no se cuenta con la infraestructura necesaria como para llevar a cabo la distribución (edificios y depósitos dañados, rutas intransitables, aeropuertos inutilizables) ni tampoco se cuenta con la suficiente cantidad de recursos (vehículos, personal calificado, etc.) [37]. Existen numerosos modelos que resuelven el problema de la distribución planteado anteriormente, en el documento Estado del Arte [Estado del Arte - punto 4.1] se tratan y describen varios de ellos.

2.3.3 EVACUACIONES

Es necesario implementar planes de evacuaciones para las zonas donde exista el riesgo de que se produzcan desastres; de esta manera, las personas que allí residan contarán con procedimientos bien definidos y sabrán cómo actuar para ponerse a salvo ante la ocurrencia del desastre [41].

Se deben establecer un conjunto de rutas de evacuación hacia los refugios o zonas seguras, de manera que la zona afectada pueda ser evacuada lo más rápido y de la forma más segura posible.

Para tal fin, es necesario investigar vulnerabilidades y riesgos existentes en la zona (a través de revisiones históricas, estudios geológicos, etc.) así como también contar con un adecuado sistema de alertas (proporcionando el tiempo necesario para ejecutar el plan de evacuación) [41].

Se deben diseñar rutas de evacuaciones principales y auxiliares, donde se busque minimizar las distancias de las mismas y el riesgo asociado a cada una de ellas (derrumbe de puentes o estructuras cercanas, rutas inundables, etc.); además se deben seleccionar como destino de la evacuación aquellos refugios o zonas cuya probabilidad de ser

afectada por el desastre sea mínimo (minimizar riesgo) [22].

2.3.4 ALBERGUES TEMPORALES

Los albergues temporales deben poder proteger a las personas afectadas, asegurando condiciones dignas y atendiendo las necesidades básicas de sus residentes, a la vez de salvaguardar la estabilidad física y psicológica de los mismos, contando con un equipo multidisciplinario compuesto principalmente por trabajadores sociales, psicólogos y doctores. [43]

Se debe contar con un adecuado saneamiento del lugar, agua potable, vestimenta, alimento e infraestructura adecuada (la Organización Mundial de la Salud plantea una guía completa con puntos a cumplirse en los albergues temporales [43]); además deben poseer una estructura segura que pueda proteger a sus residentes contra condiciones adversas, así como también estar ubicados en lugares seguros (la localización se realiza en forma individual en función de estudios de cada zona en particular) [44].

Otro punto importante en los albergues temporales es la coordinación de suministros, siendo aconsejable que se encuentren ubicados cerca de carreteras para facilitar la entrega de los mismos [44].

Existen varios tipos de albergues temporales, algunos de los más conocidos son las construcciones fijas para tal fin (instalaciones construidas específicamente para ser albergues temporales y utilizadas únicamente ante la ocurrencia de un desastre), los lugares públicos (edificios del estado, gimnasios, estadios, etc. que se utilizan provisoriamente como refugios temporales), y los campamentos de emergencia (carpas o tiendas que se levantan en zonas seguras). Muchas veces no es la severidad del desastre lo que determina qué tipo de albergue utilizar, sino que son las capacidades económicas y recursos locales los que influyen en ello [45].

En el Estado del Arte [Estado del Arte - punto 6.2] se plantean algunos modelos que son de utilidad para decidir cuáles y cuántos albergues abrir (en función de la cantidad y localización de afectados, entre otros puntos) así como también ayudan en la gestión de los mismos.

2.4 SOFTWARE PARA LA LOGÍSTICA HUMANITARIA

Actualmente existen varios sistemas que sirven de apoyo para las tareas que conciernen la logística humanitaria, siendo de gran ayuda en la toma de decisiones.

La mayoría de los sistemas se enfocan principalmente en la gestión y control de inventarios, así como en otras acciones pertenecientes a la fase de respuesta, dejando de lado tareas asociadas a la distribución y transporte de suministros.

A continuación se plantean algunos de los sistemas más importantes, conocidos y utilizados para llevar a cabo algunas de las tareas asociadas a la logística humanitaria.

2.4.1 LSS

LSS (Logistics Support System) es una herramienta de ayuda para una eficiente gestión de inventarios y manejo de suministros en un contexto de ayuda humanitaria, brindando la posibilidad de una respuesta rápida a las necesidades existentes a la vez de oportuna. Se lo considera como el sistema sucesor del conocido SUMA [50], ya que mejora sus funcionalidades a la vez que incorpora nuevas tecnologías (por ejemplo la plataforma web) [17] [19].

Proporciona información detallada de la ayuda humanitaria recibida, facilitando el control de inventario y brindando transparencia a la operación (permite crear reportes de estado que pueden ser compartidos con donantes, autoridades, medios de comunicación, etc.). A su vez, posibilita el intercambio de información de manera rápida entre los distintos agentes humanitarios y provee un esquema de coordinación que involucra los suministros que han sido prometidos así como también los que han arribado al país [48].

Otras de las principales funcionalidades del software son brindar la posibilidad de clasificar los suministros y su utilidad, especificar dónde se encuentran, definir suministros requeridos, definir el momento en el que un determinado suministro se ha enviado (encontrándose en tránsito) y especificar el momento de recepción del mismo, entre otras [17].

El sistema cuenta con dos módulos, uno de escritorio pudiéndose trabajar como monousuario o en una red local de computadores, y otro web con el que se trabaja directamente con un navegador sobre internet. A su vez, el sistema brinda la posibilidad de conferirles distintos permisos a los usuarios (funciones del sistema, como por ejemplo registrar entradas de suministros, registrar envíos de suministros y generar reportes) [48].

El módulo de escritorio opera bajo el sistema operativo Windows, pudiéndose ejecutar sobre un ambiente de red Windows compartiendo una misma base de datos centralizada [20].

El modulo web puede ejecutarse con cualquier navegador (recomendando los autores

Internet Explorer); el beneficio de que se ejecute sobre la plataforma web es que se puede subir información (por ejemplo cantidad de suministros en una bodega) al sistema en tiempo real y la misma queda visible instantáneamente para todos los usuarios, así como también realizar consultas (por ejemplo disponibilidad de determinado suministro u hora en la que salió el envío); además se permite la administración (por ejemplo de bodegas) y coordinación de envíos en forma remota [17].

2.4.2 SAHANA

Sahana Foundation ha desarrollado tres sistemas destinados a brindar asistencia en distintos ámbitos dentro del contexto de ayuda humanitaria, todos ellos son software libre y de código abierto (bajo la licencia LGPL). Estas herramientas son: Sahana Vesuvius, Sahana SEMS y Sahana Eden [21].

Sahana Vesuvius está enfocado a la comunidad médica, brindando herramientas que ayuden a asistir a estos profesionales en sus actividades y brindando una red con información de las víctimas así como también las necesidades de los hospitales o instituciones médicas que atiendan a los afectados [21].

Sahana SEMS es una herramienta que se utiliza para planear y coordinar el despliegue del personal y las instalaciones (albergues temporales) necesarias para atender a los afectados tras la ocurrencia de un desastre; a su vez, también brinda funcionalidades que permiten el registro y seguimiento de los afectados que se alojan en los albergues [21].

Finalmente, Sahana Eden, el sistema más importante y utilizado del conjunto, es una herramienta flexible, basada en la plataforma web, que puede ser utilizada en varias de las fases de la gestión de riesgos, tales como mitigación, preparación, respuesta y reconstrucción. Algunas de las funcionalidades principales del sistema son [21]:

- Registro de personas extraviadas y encontradas, teniendo la información de quiénes los buscan.
- Registro de organización, que permite saber “quién hace qué y dónde”, manteniendo registro (localización y actividades que realizan) de todas las organizaciones que se encuentren trabajando en la región del desastre; también puede servir para detectar carencias en servicios.
- Registro de albergues temporales, manteniendo información de todos los albergues abiertos (ubicación, capacidad, características, etc.) así como también de las personas que se encuentran alojadas en ellos.
- Coordinación de voluntarios, ayuda a las organizaciones a mantener un seguimiento de todos sus voluntarios (información de contacto, ubicación,

disponibilidad, capacidades) con el fin de utilizar los recursos de la mejor manera posible.

- Manejo de Inventario, lleva registro de las transacciones de suministros (desde la salida hasta la recepción del mismo), y la ubicación, cantidad, disponibilidad y clasificación de suministros. Además permite administrar las solicitudes de suministros (realizando un seguimiento de las peticiones hasta su cumplimiento), y manejo de donaciones.
- Registro de bienes, con información sobre los distintos bienes a utilizarse o utilizados en el proceso (vehículos, generadores eléctricos, equipos de comunicación, etc.) como ser características, ubicación, estado, persona responsable, etc.

2.4.3 HELIOS

Desarrollado por el instituto Fritz como sucesor del software HLS (Humanitarian Logistic System), HELIOS es una herramienta desarrollada específicamente para la gestión y coordinación de suministros en un contexto de logística humanitaria.

El sistema funciona sobre la plataforma web, desarrollado en Microsoft .Net y utiliza la base de datos Microsoft SQL Server. [22]

Permite realizar el seguimiento de los suministros a través de todo el proceso (desde el pedido del suministro hasta la recepción del mismo), a la vez de disponer de información y generar reportes acerca de la recepción de donaciones (monetarias y en productos); todo esto le brinda transparencia y seriedad a la operación. Por lo tanto, se logra una respuesta rápida y oportuna para el envío de suministros, mejorando el manejo de los mismos (almacenamiento, demanda, disponibilidad, ubicación, etc.), y a su vez disminuyendo el esfuerzo y costo (por ejemplo mejor utilización de recursos) asociado a la gestión de suministros [22].

2.4.4 HFOSS

El proyecto Humanitarian FOSS (HFOSS) es un proyecto de software libre que ha desarrollado un conjunto de aplicaciones útiles para la gestión de desastres. Dos de las más importantes son Collabit y Posit.

Collabit es un software web que sirve para el intercambio de información entre las distintas organizaciones durante la emergencia, brindando una manera rápida y efectiva de coordinar y organizar entre sí los esfuerzos asociados a la etapa de respuesta. [23]

Por su parte, Posit es una aplicación móvil desarrollada sobre la plataforma Android, utilizada principalmente para registrar determinados elementos de interés en el área, junto con una descripción, imagen y coordenadas, para luego poder visualizarlos en un mapa; por ejemplo puede servir para registrar la ubicación de albergues o lugares a evacuar. [24]

2.4.5 DMIS

DMIS (Disaster Management Information System) es un sistema web desarrollado por la Federación Internacional de la Cruz Roja para mejorar el acceso e intercambio de información entre las federaciones a nivel global, mejorando así las acciones asociadas a la preparación y respuesta ante un desastre; el sistema únicamente es accesible para miembros autorizados de la Cruz Roja. A su vez, permite el acceso a la información en tiempo real sobre las situaciones de campo en las zonas, proporcionando una percepción bastante fiel de los daños y necesidades existentes, favoreciendo una mejor planificación de la respuesta [25].

3 DEFINICIÓN DEL PROBLEMA

3.1 INTRODUCCIÓN

A continuación se describen los aspectos más importantes en el transcurso del proyecto destacando la propuesta e implementación de un prototipo.

Este proyecto consta de dos grandes tareas que son: el relevamiento de mecanismos, técnicas, definiciones, modelos, algoritmos y estudios sobre logística humanitaria, para luego, con el conocimiento adquirido, poder aplicar y adaptar alguno de ellos a la realidad de nuestro país y solucionar un problema de decisión operativa a través de una nueva herramienta de software.

Se plantea desarrollar un algoritmo o heurística que implemente el modelo para solucionar el problema de la distribución de suministros en un escenario dado por proveedores y demandantes.

Los conjuntos de proveedores y demandantes podrían ser muy grandes y puede llegar a ser muy complicado decidir a simple vista cómo y en qué orden visitar a las zonas necesitadas y cuánta ayuda humanitaria se necesita.

3.2 ANTECEDENTES

Como se menciona en la sección 4.2 del documento Estado del Arte, existe un proyecto de nombre HADS (Humanitarian Distribution System). Según las referencias, se implementó un prototipo en 2010, no estable y con funcionalidades acotadas. El prototipo está basado en un modelo de optimización multi-objetivo con restricciones lineales. Tiene varias funciones objetivo y ponderando con valores constantes a cada una de las funciones, permite darle distintas prioridades a cada objetivo. Algunos de los objetivos del modelo son:

- Satisfacer la demanda equitativamente
- Ajustar costos a un presupuesto dado
- Minimizar la probabilidad de saqueo en las rutas elegidas

A pesar de varios intentos de establecer comunicación con los autores no se pudo contar con el software para poder estudiar aspectos de funcionalidades y performance.

Algunas características del modelo fueron tomadas en cuenta para el desarrollo del prototipo de este proyecto, como por ejemplo las restricciones de capacidad y que los suministros a transportar son representados como unidades (por ejemplo una unidad puede representar un kit con determinados suministros específicos. La diferencia fundamental está en que el modelo de este proyecto toma como restricción que se satisfaga la demanda mientras que el otro lo considera como una de las funciones objetivo (opcional); el objetivo principal va a ser minimizar el tiempo de respuesta, el tiempo que transcurre hasta que los suministros llegan al destino.

Este tipo de problemas pertenece a los problemas de optimización combinatoria. Esto quiere decir que no existe un algoritmo de orden polinomial que logre encontrar la solución óptima. Se dice también que el tiempo de ejecución crece exponencialmente en función del tamaño de los parámetros de entrada, la cardinalidad del espacio de búsqueda suele ser demasiado grande como para resolver el problema con algoritmos exactos (demasiado tiempo necesario). De esta forma, se hace necesaria la búsqueda de algoritmos que permitan resolver dichos problemas de manera eficiente, obteniendo una solución de calidad en un tiempo razonable. Para tal fin, se utilizan heurísticas y metaheurísticas que resuelvan los problemas de ruteo de vehículos.

Hasta donde se ha podido indagar, no se encuentran antecedentes que abarquen la resolución del problema de distribución de suministros en situación de emergencia en Uruguay.

3.3 HIPÓTESIS O SUPUESTOS

A la hora de utilizar el sistema, deben darse ciertas condiciones para que sea útil la respuesta del mismo.

Como primera condición, debe ser conocido cada uno de los lugares en donde hay personas que necesitan suministros y poder ubicarlos en un mapa georeferenciado. De la misma forma, debe ser conocida la ubicación de los centros donde están los suministros a distribuir.

Debe hacerse algún tipo de relevamiento para poderse estimar tanto la cantidad de insumos necesarios por las personas afectadas como la cantidad de suministros disponibles en los lugares de carga. Es muy importante que este relevamiento se realice de la mejor forma posible, ya que el éxito de la acción de respuesta humanitaria, es decir, que las zonas afectadas sean atendidas de la mejor manera posible, depende

fuertemente de entender cuáles son sus necesidades.

3.4 RESPUESTA RÁPIDA

El objetivo principal de la implementación del prototipo, es brindar la posibilidad de ahorrar tiempo en la etapa de respuesta en el escenario de una emergencia, obteniendo la forma en que se distribuyen los suministros en un corto periodo de tiempo, tomando en cuenta como función objetivo a minimizar el tiempo en lugar de los costos como cualquier otro sistema de ruteo.

No se debe perder de vista que el aspecto más importante en logística humanitaria es minimizar el sufrimiento de las víctimas, no siendo relevantes (en un principio) los costos asociados.

Para ello se implementó un sistema que utiliza un componente visual para mostrar los resultados en un mapa y una componente lógica que realiza los cálculos para llegar a la solución. El sistema pretende resolver en poco tiempo la distribución de todos los suministros y garantizar que la distribución sea optima en términos del tiempo transcurrido desde que los vehículos parten desde el lugar de carga hasta que los vehículos llegan a la zona afectada donde se necesitan insumos.

3.5 PROBLEMA A RESOLVER

Definimos entonces, el problema a resolver, como la suma de la investigación realizada y el prototipo de software desarrollado que intenta aplicar algunos de los conocimientos adquiridos en el transcurso del proyecto, incluyendo, un nuevo modelo de distribución que surge a partir del modelo descrito en la sección 8.1, y la construcción de un algoritmo que lo implementa brindando una solución aproximada.

El objetivo de la investigación es el resumen de temas relacionados a logística humanitaria, diferencias entre logística humanitaria y logística comercial, y descripción de los modelos y algoritmos más relevantes dentro de los encontrados. El resultado es el documento de estado del arte para logística humanitaria.

El prototipo de software, tiene como objetivo principal resolver el problema de ruteo de vehículos en una situación de emergencia en Uruguay. Se propone implementar un sistema de información para acelerar el proceso de toma de decisiones por parte de quienes son responsables de la distribución de suministros.

Los principales elementos que componen el sistema son:

- Mapa, se denomina mapa a una instancia del modelo que funciona como contenedor de nodos. Se puede especificar la cantidad de vehículos disponibles para cada instancia de Mapa.
- Nodos proveedores, tienen como propiedades su ubicación geográfica, cantidad de suministros disponibles, cantidad de vehículos y nombre que puede ser asociado a su localidad o algún otro criterio que sea necesario.
- Nodos demandantes, tienen como propiedades su ubicación geográfica y cantidad de suministros necesarios en la zona.
- Trayectos, son representados con líneas que unen nodos (demandantes, proveedores o ambos) y cada trayecto tiene la cantidad de suministros transportada entre sus extremos.
- El objetivo que busca optimizar el sistema es minimizar el tiempo necesario para realizar la distribución de los suministros de los nodos Proveedores hacia los Demandantes, de manera que los suministros lleguen lo más rápido posible a la población afectada.

Este sistema permite resolver el problema de distribución de suministros y o personas desde ciertos puntos de partida hacia donde son necesitados.

La salida de la función principal del sistema es una distribución que tiene cuáles destinos se visitan, desde que orígenes, en qué orden y cuantas unidades de suministros es transportado en cada tramo y por consecuencia, cuántas unidades se dejan en cada sitio.

Para desplegar los resultados, se implementó en la interfaz de usuario un módulo que contiene un mapa. Los nodos se posicionan en el lugar correcto del mapa permitiendo así que la solución sea fácilmente visualizada y la interacción entre los usuarios y el sistema sea amigable.

Debido al acotado tiempo de desarrollo no se pudieron implementar todos los aspectos que se deseaban desarrollar en un principio. Era de gran interés por ejemplo, incluir en el modelo la posibilidad de tratar a los suministros como productos individuales (existiendo tantos tipos de suministros en el sistema como se deseara). Para esto se pensó en desarrollar un módulo que permitiese gestionar y mantener un registro de los mismos (por ejemplo altas, bajas y modificaciones de suministros). Como ejemplo se podría Indicar el origen de cada suministro y luego para qué y cuándo fue utilizado, mejorando así la transparencia de toda la gestión.

Otra funcionalidad deseable y muy importante que quedó fuera del alcance del proyecto es la exportación de la solución, es decir que la información generada por el sistema debe ser exportada a planillas de cálculo o algún formato adecuado y expresarla de

forma analítica para mejorar el registro y portabilidad de la misma.

4 DESCRIPCIÓN DE LA SOLUCIÓN

4.1 ALCANCE

El sistema desarrollado básicamente consiste en una aplicación web que facilita la toma de decisiones para la distribución de suministros entre un conjunto de nodos dispuestos en un mapa, logrando en un tiempo acotado (en lo posible unos pocos segundos) una distribución lo más eficiente posible.

Para ubicar los nodos y visualizar la solución el usuario cuenta con el mapa de Google [52]. De esta manera, se saca provecho de los beneficios y funcionalidades que nos brinda nativamente los mapas de Google, como por ejemplo una cartografía urbana muy completa a nivel mundial, capacidad de vista en modo satélite, relieve de la superficie, la función Street View, etc.

Existen dos tipos de nodos en el sistema, los nodos Demandantes y los nodos Proveedores:

- Los Demandantes son los nodos a los que habrá que abastecer con la cantidad de suministros que demanden (o intentar abastecer en la mayor medida posible).
- Por su parte, los nodos Proveedores serán los nodos que posean suministros en el sistema.

De esta manera, cada vez que un usuario quiera agregar un nodo al mapa, deberá poder especificar el nombre del nodo (típicamente el nombre del pueblo, zona o ciudad a la que representa), el tipo de nodo (Proveedor o Demandante) y la cantidad de suministros que tiene para distribuir o que demanda (dependiendo del tipo del nodo).

Vale aclarar, que los suministros se toman como unidades, por ejemplo, una unidad de suministro puede corresponder a una canasta básica o un paquete con determinados alimentos/insumos.

Por otra parte, en el modelo tomado como referencia [Estado del Arte – punto 4.2], se especifica que cada nodo de tipo Proveedor posee vehículos con los cuales realiza la distribución. A su vez, también es posible especificar la capacidad exacta de dichos vehículos (en unidades de suministros); por lo tanto, el usuario también podrá especificar la capacidad de los vehículos (asociados a un mapa) en cualquier momento.

Una vez se haya definido un mapa con todos sus elementos (nodos Proveedores y

Demandantes, capacidad de los vehículos, etc.), se puede calcular la distribución solución.

Como se mencionó anteriormente, cada nodo Proveedor dispone de una cierta cantidad de vehículos para realizar su distribución, por lo tanto, la solución estará compuesta por cada vehículo saliendo desde cada nodo Proveedor, visitando determinados nodos Demandantes, y finalmente volviendo nuevamente al nodo Proveedor. A su vez, en la solución se indica la cantidad de suministros que debe llevar cada vehículo, el orden en los cuales debe visitar cada nodo Demandante, la ruta (en el mapa) que debe seguir, y la cantidad de suministros que se deben dejar en cada nodo Demandante.

Es importante mencionar que una vez que se calcula una solución, la cantidad de suministros, tanto en los nodos Demandantes como en los Proveedores, varía actualizándose según la cantidad de demanda satisfecha y la cantidad de suministros utilizados en los nodos Demandantes y Proveedores respectivamente, considerando la solución calculada anteriormente.

Además, es posible ir actualizando los nodos en función de los nuevos suministros que lleguen a los nodos Proveedores y las nuevas demandas existentes en los nodos Demandantes, de manera de poder calcular nuevamente la distribución de los nuevos recursos considerando a su vez las nuevas demandas. Finalmente, el usuario también podrá crear nuevos mapas, guardarlos y cargarlos.

También, es deseable que el sistema cuente con la posibilidad de mostrar determinados datos de interés, como ser zonas proclives o con antecedentes de inundaciones, incendios, etc., por ejemplo utilizando servicios WMS (Web Map Service); el SINAE brinda de manera gratuita una serie de geoservicios web con este tipo de información.

4.2 REQUERIMIENTOS

El requerimiento principal del sistema desarrollado es proveerle al usuario ayuda al momento de planificar la distribución entre un conjunto de proveedores hacia un conjunto de clientes distribuidos geográficamente en el mapa. De esta manera, se busca obtener una solución que plantee una distribución eficiente y a su vez poder presentársela al usuario de forma clara, de manera que resulte fácil para el usuario entender la distribución presentada, y así poder llevarla a cabo.

Para tal fin, será necesario integrar un mapa georeferenciado a la interfaz de usuario, de manera que el usuario pueda interactuar con el mismo (por ejemplo agregando nodos o visualizando la distribución solución).

A continuación se plantean los requerimientos funcionales y no funcionales que creemos debería necesitar un sistema para cumplir con el alcance planteado [punto 4.1] así como también con los puntos mencionados anteriormente.

4.2.1 FUNCIONALES

- Se debe de poder crear, guardar y cargar mapas en el sistema.
- Al crear un mapa, se debe poder especificar la capacidad de los vehículos utilizados en el mismo (todos los vehículos de un mapa tienen la misma capacidad); a su vez, dicha capacidad debe poder modificarse luego.
- Deben existir dos tipos de nodos en el sistema: proveedores y demandantes. Los nodos proveedores serán los que contengan los suministros a distribuirse en la operación, a la vez que contienen vehículos (con capacidad) para realizar la distribución. Los nodos demandantes tendrán asignada una demanda.
- Se deben poder dar de alta, modificar y eliminar nodos en un mapa. De los nodos proveedores se debe poder modificar la cantidad de suministros que tiene, y de los demandantes se debe poder modificar la cantidad de suministros demandados.
- Los nodos deben estar referenciados geográficamente, de manera que cada nodo se corresponda con una coordenada en el mapa. La ubicación de un nodo en el mapa debe poder especificarse mediante un click en el mapa.
- Se debe poder calcular y mostrar una distribución solución para cada mapa.
- Una solución debe indicar la cantidad de suministros utilizados de cada proveedor, así como también la ruta que deben seguir los vehículos asociados a cada uno de los proveedores (distribución) y la cantidad de suministros que quedan en cada demandante.
- La distribución solución debe representarse gráficamente sobre el mapa.
- Una vez que se calcula una solución se deben de actualizar, en función de los suministros transportados, la cantidad de suministros que se encuentran en los nodos proveedores así como también la cantidad de suministros demandados por los nodos demandantes.

- Brindar la posibilidad de poder visualizar en el mapa capas con los datos de los servicios WMS proporcionados por el SINAIE.

4.2.2 NO FUNCIONALES

- El tiempo necesario para calcular y representar la distribución solución debe ser acotado (en la medida de lo posible unos pocos segundos).
- La distribución solución calculada debe buscar minimizar los tiempos necesarios para realizar las entregas.
- La distribución solución presentada al usuario debe ser clara y fácilmente entendible por el mismo.
- Se garantiza el correcto funcionamiento del sistema sobre los navegadores Mozilla Firefox y Google Chrome.
- Se debe desarrollar la aplicación utilizando software libre, de manera de que cualquier persona luego pueda ver y modificar el código sin necesidad de contar con licencias o programas pagos.

4.3 MODELO DE DISTRIBUCIÓN

El modelo de distribución que se utiliza en la solución, se obtiene a partir del estudiado en el documento Estado del Arte [Estado del Arte – punto 4.2].

Es importante mencionar que para la resolución del modelo se utiliza el solver GLPK[18], a su vez, también se desarrolló una heurística que resuelve el mismo problema planteado en el modelo. Lo que se logra con esto es que en caso de que la cantidad de datos sea reducida se obtiene la solución óptima utilizando el solver, y si la cantidad de datos aumenta (o el tiempo de computo necesario para resolverlo con el solver es demasiado) se utiliza la heurística. Además, es importante mencionar que en el modelo planteado (resuelto utilizando el solver GLPK como ya mencionamos), solamente se encontrará una solución si se logra satisfacer la totalidad de la demanda, en caso contrario se recurre a la heurística.

Considerando lo anterior, lo que se hizo a la hora de desarrollar el modelo que se plantea a continuación, básicamente fue simplificar y modificar algunos de los principios planteados en el modelo desarrollado por Ortuño et al. [28] [Estado del Arte – punto 4.2]. De esta manera, se eliminaron algunos parámetros, variables y restricciones que no aportan demasiado a la resolución del problema y lo hacen ganar en complejidad, a la vez de modificar algunas variables y restricciones. Por ejemplo se eliminaron parámetros referentes a la confiabilidad y probabilidad de saqueo asociada a cada camino del mapa, también se eliminaron parámetros correspondientes a la velocidad que se puede desarrollar en cada camino así como también costos fijos y de reparación asociados a los vehículos (se elimina la cota asociada al presupuesto).

La eliminación de todos estos parámetros, lógicamente también implica eliminar y modificar muchas de las restricciones originales.

Finalmente, estas modificaciones también repercutieron en los tiempos de cómputo necesarios para resolver el problema (en el punto 5.2.1 se describen las pruebas de performance sobre el modelo actual en función del número de nodos, utilizando el solver GLPK).

Además, el desarrollo de una heurística que resolviera el problema planteado en el modelo original [28] nos pareció en exceso complejo, excediendo el alcance propuesto para el proyecto de grado.

Vale aclarar que en un principio, también se desarrolló otro modelo (nuevamente basándonos en el modelo [28]) que agrega algunos atributos de interés (por ejemplo se trabajan los suministros como elementos individuales, y no como un conjunto), acercando el modelo aún más a la realidad; claro que esto también hace aumentar la complejidad del mismo, y por ende aumentar los tiempos de cómputo; en el Anexo [punto 8.1] se plantea, desarrolla y explica en detalle el mismo.

El motivo principal por el cual no se utiliza este modelo a efectos prácticos en el proyecto es porque los tiempos de cómputo necesarios para obtener una solución utilizando un solver son extremadamente altos; se modeló el mismo con el GLPK y realizaron algunas pruebas con diferentes cantidades de nodos y datos (como ser número de vehículos, suministros, etc.) obteniéndose tiempos de resolución demasiados elevados a los intereses del proyecto (uno de los objetivos es obtener soluciones de manera veloz). Además, desarrollar una heurística que resolviera el modelo planteado, excedía el alcance propuesto para el proyecto.

A continuación se procederá a desarrollar el modelo utilizado en el proyecto.

El problema que intenta resolver el modelo es calcular las rutas para la distribución de suministros (de manera de satisfacer la demanda), desde los nodos proveedores a los nodos demandantes, siendo que las capacidades de los vehículos son las mismas y cada proveedor puede tener distinta cantidad de vehículos (generándose un camino para

cada uno de los vehículos de cada nodo proveedor). A su vez, como el objetivo del problema es atender lo antes posible a cada uno de los clientes existentes (sin considerar costos asociados), el tiempo que se busca minimizar es el que corresponde a las rutas solución de la distribución.

A continuación definiremos los datos asociados al modelo:

- p_i : nodos proveedores, con $i \in \{1, \dots, n\}$. Al conjunto de nodos proveedores los llamaremos P .
- d_i : nodos demandantes, con $i \in \{1, \dots, m\}$. Al conjunto de nodos demandantes los llamaremos D .
- a_i : conjunto de enteros positivos, con $i \in P$, que representan la cantidad de vehículos asociados a cada uno de los nodos proveedores p_i .
- t_{ij} : tiempo (provisto por la API de Google Maps [52]) requerido para ir desde el nodo i al nodo j , con $i, j \in D \cup P$; t_{ii} se define como 0.
- sum_i : cantidad de unidades de suministros, con $i \in P$, correspondientes al proveedor p_i .
- dem_i : cantidad de unidades de suministros demandadas, con $i \in D$, correspondientes al demandante d_i .
- e : capacidad de los vehículos en unidades de suministros.

Variables de decisión asociadas al modelo:

- $LORRY_{ij}$: número de vehículos utilizados en el camino desde el nodo i al nodo j ; con $i, j \in D \cup P$.
- $LOADTR_{ij}$: cantidad de mercadería llevada desde i a j ; con $i, j \in D \cup P$.
- $LOAD_i$: cantidad de mercadería que se encuentra en el nodo i al final de la operación; con $i \in D \cup P$.

Las restricciones del modelo vienen dadas por:

$$LOADTR_{ij} = 0, \quad \forall i \in D \cup P, \quad \forall j \in P \quad (1)$$

Especifica que no se debe llevar mercadería hacia los nodos proveedores; de no existir esta restricción podría darse el caso de sobrecargar los vehículos y que los mismos cuando regresen al nodo proveedor desde donde partieron regresen con suministros.

$$\sum_{i \in D, t_{ji} > 0} LOADTR_{ji} = sum_j - LOAD_j, \quad \forall j \in P \quad (2)$$

Define la correspondencia de carga para los nodos proveedores; por ejemplo que no puedan salir más suministros desde el nodo proveedor de los que éste realmente tiene.

$$\sum_{i \in D, t_{ij} > 0} LOADTR_{ij} - \sum_{i \in D, t_{ji} > 0} LOADTR_{ji} = LOAD_j, \quad \forall j \in D \quad (3)$$

Define la correspondencia de carga para los nodos demandantes.

$$LOAD_i = dem_i, \quad \forall i \in D \quad (4)$$

Se debe cumplir con la totalidad de la demanda existente.

$$LOAD_i \leq sum_i, \quad \forall i \in P \quad (5)$$

Los nodos proveedores no deben recibir suministros, a la vez de no

transportan los suministros que no sean necesarios distribuir.

$$\sum_{j \in DUP, t_{ij} > 0} LORRY_{ij} \leq a_i, \forall i \in P \quad (6)$$

Especifica que no pueden salir más vehículos del nodo proveedor de los que éste realmente tiene.

$$\sum_{i \in DUP, t_{ij} > 0} LORRY_{ij} - \sum_{i \in DUP, t_{ji} > 0} LORRY_{ji} = 0, \forall j \in D \cup P \quad (7)$$

Define la correspondencia de vehículos en cada nodo, de manera que no queden vehículos en los nodos demandantes que se visitan, debiendo regresar a los nodos proveedores.

$$LOADTR_{ij} \leq e \cdot LORRY_{ij}, \forall i, j \in D \cup P, t_{ij} > 0 \quad (8)$$

Define el límite de carga transportada en un camino como la suma de las capacidades de los vehículos que por el transitan (recordar que todos los vehículos tienen la misma capacidad).

$$e \cdot LORRY_{ij} - e \leq LOADTR_{ij}, \forall i \in D \cup P, \forall j \in D, t_{ij} > 0 \quad (9)$$

Fuerza a no utilizar vehículos innecesarios (vacíos) para llevar suministros a los demandantes; en el caso de los vehículos que viajan de vuelta hacia los nodos proveedores (desde los últimos nodos demandantes visitados) si pueden (y deben) venir vacíos.

Finalmente, la función objetivo es:

$$\min \sum_{i \in DUP} \left(\sum_{j \in DUP} (t_{ij} \cdot LORRY_{ij}) \right)$$

Por lo tanto, el modelo planteado resulta de la siguiente forma:

$$\begin{aligned} \min & \sum_{i \in DUP} \left(\sum_{j \in DUP} (t_{ij} \cdot LORRY_{ij}) \right) \\ \text{s. t.} & \\ & (1) - (9) \\ & LORRY_{ij}, LOADTR_{ij}, LOAD_i \geq 0 \text{ y enteros} \end{aligned}$$

4.4 DISEÑO Y FUNCIONAMIENTO DE LA HEURÍSTICA

La heurística a desarrollarse debería poder resolver el modelo planteado en el punto anterior, eso es a grandes rasgos, calcular la distribución para un conjunto de proveedores, con determinadas cantidades de suministros y determinada cantidad de vehículos asociado a cada proveedor (capacidad uniforme entre los vehículos) y un conjunto de clientes con determinadas demandas, siempre buscando atender a cada uno de los clientes lo antes posible (reducir los tiempos asociados a la distribución sin considerar la ruta de vuelta desde el ultimo cliente hacia el proveedor).

Si bien el solver GLPK permite resolver el modelo planteado en el punto anterior brindando la distribución optima, el tiempo de computo necesario para obtenerla es demasiado alto (considerando los objetivos del proyecto), así como también la cantidad de nodos que puede manejar el solver es demasiado acotada; pruebas realizadas sobre el modelo utilizando el GLPK pueden observarse en el punto 5.2.1.

Sin embargo, es importante mencionar que si bien el sistema generalmente no utilizará el solver para obtener la distribución solución, en los casos en que el tiempo de ejecución para el cálculo de la misma sean menor a 2 segundos sí se utilizara el solver (el funcionamiento específico del sistema en función de la utilización del solver/heurística se define en el punto 4.6.2.1)

En el resto de los casos (cuando no se utilice el solver), se utilizara la heurística que se desarrolla en el presente punto, brindando tiempos de resolución acotados, permitiendo definir escenarios con gran cantidad de nodos, y a su vez ofreciendo soluciones de calidad (en el punto 5.2.2 pueden observarse pruebas realizadas sobre la heurística desarrollada)

En el documento Estado del Arte se estudian algunas heurísticas y metahurísticas existentes para resolver problemas de ruteo [Estado del Arte - punto 7].

Una de las heurísticas más conocidas para el problema de CVRP (Capacitated Vehicle Routing Problem) es el Algoritmo de Ahorros de Clarke y Wright [55] [56], el cual consiste en partir de una solución inicial e ir combinando las rutas existentes para buscar obtener mejoras en el costo total (generalmente se utilizan operadores de búsqueda local para mejorar la solución obtenida con este algoritmo).

Otra de las heurísticas conocidas para los problemas de ruteo es la desarrollada por Solomon [57], la cual podría considerarse como una extensión del algoritmo de ahorros de Clarke y Wright para solucionar problemas con ventanas de tiempo.

A su vez, también pueden encontrarse algunas heurísticas en la web (desarrolladas en artículos de posgrado, maestrías o documentos científicos) para resolver problemas de ruteo con diferentes características (generalmente desarrolladas para un problema específico con determinadas características específicas).

Si bien, como ya se mencionó, existen varias heurísticas que intentan resolver problemas de ruteo, algunas de las características particulares del problema que se intenta resolver en el proyecto actual hacen que su aplicabilidad no sea directa, debiéndose adaptar y desarrollar una nueva heurística sobre algunas de las tradicionales ya existentes.

Por ejemplo, las heurísticas de Clarke – Wright y Solomon mencionadas anteriormente, resuelven el problema de ruteo para un solo proveedor y un conjunto de clientes, pero cuando se agregan varios proveedores al sistema el problema cambia sensiblemente, ya

que los proveedores deben “interactuar” entre sí para distribuirse los clientes existentes. Además, ambas heurísticas (al igual que la mayoría) buscan reducir la distancia recorrida en la solución (y por ende el coste asociado), por lo que buscan reducir la cantidad de vehículos que se utilizan en la solución (buscando combinar caminos si es posible), lo cual no siempre es conveniente si se quiere atender a cada uno de los clientes lo antes posible (con varios vehículos se logra llegar antes a los nodos, probablemente ganando en costo y distancia).

A su vez, muchas de las heurísticas encontradas en la web son heurísticas desarrolladas sobre alguna heurística clásica y adaptada para resolver el problema específico que trate el artículo y/o documento presentado.

Por lo tanto, se procedió a desarrollar una heurística propia basándonos fuertemente en la heurística de Cluster first – Route second y la heurística de Inserción (ambas estudiadas en el Estado del Arte - punto 7).

De esta manera, para realizar una zonificación donde cada proveedor resulte asociado a un conjunto de clientes (a los cuales debe abastecer), nos basamos en la heurística Cluster first – Route second [Estado del Arte - punto 7.2.1.3], de manera de crear un cluster con clientes para cada proveedor. Luego, en una segunda instancia, dentro de cada zona se utiliza una heurística de Inserción [Estado del Arte - punto 7.2.1.2] (caracterizada por obtener soluciones rápidas, sin que la misma pierda demasiada calidad) para generar el ruteo de cada uno de los vehículos pertenecientes al proveedor para abastecer a los clientes asociados al mismo.

En primer instancia, para asignar los clientes a los proveedores, se crea una lista ordenada de clientes según la distancia de cada uno de ellos al proveedor más cercano (en primer lugar se encuentra el cliente que se halle más cercano a alguno de los proveedores del sistema). Se utiliza una idea similar a la que emplean Franco e Islas en su trabajo [29], donde para cada cliente se calcula el costo de asignarlo a cada uno de los proveedores y luego se realiza la asignación cliente-proveedor en función de dicho costo.

Luego, se va recorriendo la lista y asignando los clientes a cada uno de los proveedores; se comienza por el cliente ubicado al principio de la lista (cliente con menor distancia), se intenta asignar dicho cliente al proveedor correspondiente en la lista, luego se hace lo mismo con el segundo cliente en la lista y así sucesivamente.

Vale aclarar que al momento de asignar un cliente a un proveedor se debe tener en cuenta la cantidad de suministros que le quedan a este último y también la capacidad que le van quedando a los vehículos asociados a dicho proveedor (estos valores se van actualizando a medida que se asocian los clientes a los proveedores).

Básicamente, lo que se hace es asociar a cada cliente con el proveedor más cercano que pueda abastecer su demanda (o parte de ella); y en caso de que el proveedor no pueda

“atender” a dicho cliente, el mismo intentará ser atendido por el siguiente proveedor más cercano a él (esta lógica se continúa hasta que el cliente consiga un proveedor que pueda satisfacer su demanda o hasta que no hayan más proveedores).

Otra idea que se manejó al momento de realizar la asignación de los clientes con los proveedores fue la de ir asignando el cliente más cercano a cada proveedor, alternando entre los proveedores. De esta manera, se elige un proveedor y se le asigna el cliente más cercano a él (siempre que se respeten las restricciones del problema, como ser cantidad de suministros del proveedor y capacidad del vehículo), luego se realiza lo mismo con cada uno de los proveedores, formando una lista circular con los proveedores donde en cada iteración se le asigna al proveedor el cliente que se encuentre más cercano a él.

Este tipo de asignación tiene buenos resultados cuando los clientes se encuentran dispersos entre los proveedores; pero cuando los clientes se encuentran concentrados cerca de solamente algunos proveedores los resultados obtenidos tienden a ser malos ya que los clientes tienden a asignarse de manera uniforme entre los proveedores en vez de asignarlos a aquellos más cercanos.

A continuación se plantea el pseudocódigo del algoritmo empleado, con explicaciones, para realizar la asignación de los clientes con los proveedores:

```

Para cada elemento  $x_i$  de la lista ordenada de <cliente-proveedor> hacer {
    Sea  $c_i$  el cliente asociado al elemento  $x_i$ 
    Sea  $dc_i$  la cantidad de demanda insatisfecha del cliente  $c_i$ 
    Sea  $p_j$  el proveedor asociado al elemento  $x_i$ 
    Sea  $sp_j$  la cantidad de suministros que aun le quedan al proveedor  $p_j$ 
    Sea  $cap_j$  la capacidad total que le queda al conjunto de vehículos
    asociados al proveedor  $p_j$  (capacidad_vehículo *
    cantidad_vehículos_proveedor)

    Si ( $sp_j > 0$  y  $cap_j > 0$ ) { [el proveedor tiene suministros y sus vehículos
                                todavía no están llenos]
        Si ( $sp_j \geq dc_i$ ) { [la cantidad de suministros del proveedor dan
                            para satisfacer la demanda del cliente]
            Si ( $cap_j \geq dc_i$ ) { [la capacidad de los vehículos dan para
                                llevar toda la demanda del cliente]
                 $sp_j = sp_j - dc_i$ 
                 $cap_j = cap_j - dc_i$ 
                 $dc_i = 0$ 
                "Saco de la lista de tuplas cliente-proveedor a  $c_i$  y
                 se lo asigno al proveedor  $p_j$ "
            }
        }
    Sino { [no da la capacidad de los vehículos para
            llevar toda la demanda del cliente; solo puedo
            llevar lo que me queda de capacidad en la
            variable  $cap_j$  (capacidad_vehículo *
            cantidad_vehículos_proveedor)]
         $sp_j = sp_j - cap_j$ 
         $cap_j = 0$  [los vehículos no tienen más capacidad
                    para cargar suministros]
    }
}

```

```

         $dc_i = dc_i - cap_j$ 
        "Saco al proveedor  $p_j$  de los posibles proveedores a
        distribuir suministros (porque ya no puede distribuir
        más suministros). Por lo tanto, para cada cliente que
        esté en la lista de cliente-proveedor y que tenga como
        proveedor a  $p_j$ , calculo su distancia al próximo
        proveedor más cercano; luego reordeno esa lista de
        cliente-proveedor ( $c_i$  sigue en esa lista pero su
        demanda insatisfecha se actualizó)"
    }
}
Sino {
    [el proveedor no puede satisfacer toda la demanda del
    cliente]
    Si ( $cap_j \geq sp_j$ ) {
        [llevo todos los suministros que tiene
        el proveedor en los vehículos]
         $sp_j = 0$ 
        [el proveedor ya no tiene más
        suministros]
         $cap_j = cap_j - sp_j$ 
         $dc_i = dc_i - sp_j$ 
        "Saco al proveedor  $p_j$  de los posibles proveedores a
        distribuir suministros (porque ya no puede distribuir
        más suministros). Por lo tanto, para cada cliente que
        esté en la lista de cliente-proveedor y que tenga
        como proveedor a  $p_j$ , calculo su distancia al próximo
        proveedor más cercano; luego reordeno esa lista de
        cliente-proveedor ( $c_i$  sigue en esa lista pero su
        demanda insatisfecha se actualizó)"
    }
    Sino { [los vehículos no pueden llevar todos los suministros
    que tiene el proveedor, llevo todo lo que puedan
    llevar los vehículos]
         $sp_j = sp_j - cap_j$ 
         $cap_j = 0$ 
         $dc_i = dc_i - cap_j$ 
        "Saco al proveedor  $p_j$  de los posibles proveedores a
        distribuir suministros (porque ya no puede distribuir
        más suministros). Por lo tanto, para cada cliente que
        esté en la lista de cliente-proveedor y que tenga
        como proveedor a  $p_j$ , calculo su distancia al próximo
        proveedor más cercano; luego reordeno esa lista de
        cliente-proveedor ( $c_i$  sigue en esa lista pero su
        demanda insatisfecha se actualizó)"
    }
}
}
Sino {
    [el proveedor no tiene suministros o sus vehículos no tienen
    más capacidad]
    "Saco al proveedor  $p_j$  de los posibles proveedores a distribuir
    suministros (porque ya no puede distribuir más suministros).
    Para cada cliente que este en la lista de cliente-proveedor y que
    tenga como proveedor a  $p_j$ , calculo su distancia al próximo
    proveedor más cercano; luego reordeno esa lista de cliente-
    proveedor"
}
}

```

Cabe mencionar que, por la manera en la que se resuelve la asignación de los clientes con los proveedores, pueden suceder situaciones en las que un cliente se encuentre

asociado a dos o más proveedores, donde cada uno de los proveedores satisface una porción de la demanda del cliente.

Una vez que se tienen asociados los clientes a los proveedores se procede a realizar el cálculo de las rutas en cada proveedor.

Para realizar el cálculo de las rutas en cada proveedor (una por cada vehículo que se utilice) se aplica una heurística de inserción para cada uno de los vehículos utilizados, donde el siguiente nodo a insertarse se decide en función de la distancia al último nodo agregado (nodo más cercano).

La lógica que se sigue en cada iteración es visitar el próximo nodo más cercano que aun tenga demanda insatisfecha (cada vez que un vehículo visita un nodo actualiza el valor de demanda del mismo). Por lo tanto, en cada iteración se busca obtener la menor distancia “nodo donde esta vehículo (último nodo visitado)” – “nuevo nodo demandante a visitar” de entre el conjunto de nodos donde se encuentran los vehículos y los nodos demandantes que aún restan visitar; en su defecto vuelva al proveedor.

Finalmente cuando se han visitado todos los clientes asociados al proveedor o no existe más demanda insatisfecha, se vuelve al proveedor.

Otra manera de resolver este mismo problema de enrutamiento dentro de una zona podría ser el alternar entre los diferentes vehículos de un proveedor, eligiendo en cada iteración el nodo más cercano al vehículo correspondiente a dicha iteración. Uno de los problemas que aquí suceden es que puede darse que el ultimo nodo visitado sea atendido por un vehículo que no sea el más cercano al mismo, sino el que corresponde al vehículo seleccionado en ese estado de la iteración.

Una alternativa podría ser el no alternar entre los diferentes vehículos de un proveedor, el problema que tiene esta última metodología consiste en que los primeros vehículos visitarán los demandantes más cercanos, mientras que los últimos deberán visitar los más lejanos. De esta manera, en comparación con la metodología anterior, los demandantes más cercanos serán atendidos apenas más rápido mientras que los más lejanos sensiblemente más tarde.

A nuestro entender, la heurística planteada nos proporciona una solución de calidad aceptable en un tiempo de computo relativamente bajo; igualmente una manera de mejorar la solución podría ser aplicarle un algoritmo de búsqueda local 2-opt como el que se describe en [1], teniendo como consecuencia un aumento significativo en el tiempo necesario de cómputo.

4.5 HERRAMIENTAS UTILIZADAS

A continuación se mencionan las herramientas utilizadas para la implementación del software desarrollado; a su vez, se argumenta la elección de cada una de ellas.

4.5.1 JAVA

Para la implementación del proyecto se decidió por utilizar el lenguaje Java, debido a que ambos integrantes del equipo ya teníamos cierta experiencia con el mismo, acelerando así la curva de aprendizaje.

Otro de los puntos influyentes a la hora de la elección del lenguaje fueron las ventajas que nos otorga el mismo, algunas de las cuales son [2] [3]:

- Es un lenguaje open source (código abierto), por lo que no es necesario el pago de patentes para trabajar con el mismo
- Es Multiplataforma, por lo que funciona en cualquier Sistema Operativo para el cual exista una JVM (Java Virtual Machine).
- Lenguaje orientado a Objetos (brindando modularidad y reutilización de código).
- Administra la memoria de manera automática (garbage collector).
- Lenguaje semicompilado, combinando la rapidez y eficiencia de los lenguajes compilados con la universalidad de los lenguajes interpretados.
- Seguridad brindada por la JVM y por la API de Java.
- Gran cantidad de información en la web.
- Etc.

4.5.1.1 JDK

JDK (Java Development Kit [61]) es un conjunto de herramientas (provisto por Sun Microsystems Inc.) que nos permiten desarrollar aplicaciones en lenguaje Java. En este proyecto se utilizó la versión JDK 1.7.

4.5.1.2 NETBEANS

Netbeans [61] es un Entorno de Desarrollo Integrado (IDE), gratuito, desarrollado por Sun Microsystems principalmente para el desarrollo de aplicaciones en lenguaje Java (aunque en la actualidad existen módulos que permiten desarrollar en muchos otros lenguajes de programación).

La versión del IDE que se utilizó en este proyecto es la 7.2

4.5.1.3 JAVA EE

Java Platform Enterprise Edition (Java EE) [4] es una plataforma Java que nos permite, entre otras cosas, la creación de aplicaciones distribuidas en una arquitectura multicapa, brindándonos de manera nativa, infraestructura para el acceso a la base de datos, gestión de persistencia, administración del ciclo de vida de los objetos, control de seguridad y gestión de transacciones. [5]

La versión utilizada en el proyecto es Java EE 6.

4.5.1.4 GLASSFISH SERVER

GlassFish [60] es un servidor de aplicaciones open source, gratuito, desarrollado por Sun Microsystems que permite la ejecución de programas que sigan las especificaciones definidas por Java EE.

La versión que se utiliza en el proyecto es la 3.1.2

4.5.1.5 JDBC

Java Database Connectivity (JDBC) [6] es una API que define un estándar para el acceso a las bases de datos relacionales, además de permitir la ejecución de operaciones SQL desde la aplicación java. De esta manera, los fabricantes de bases de datos pueden crear los drivers necesarios para permitirles a las aplicaciones java el acceso a los datos. En este proyecto, al utilizarse una base de datos MySQL, se utiliza el driver “MySQL JDBC connector” para conectar a la aplicación con la base MySQL.

4.5.1.6 EJB

Los Enterprise Java Beans (EJB) [7] proporcionan un modelo de componentes que gestionan recursos (bases de datos, ficheros, etc.) y proporcionan servicios (seguridad, transacciones, concurrencia, etc.); de esta manera, son de gran utilidad a la hora de interactuar con las bases de datos, ya que liberan al programador de realizar tareas asociadas a nivel del sistema (por ejemplo seguridad y transacciones).

4.5.1.7 JPA

Java Persistence API (JPA) [8] es una API de persistencia que permite manipular y gestionar datos relacionales en aplicaciones Java. De esta manera, permite un mapeo

entre las tablas de la base (junto con sus relaciones) y objetos java.
En el proyecto, se utiliza el proveedor de persistencia EclipseLink JPA 2.0

4.5.1.8 JSP

Las JavaServer Pages (JSP) [9] permiten, a través de código java, la generación dinámica de contenido (por ejemplo HTML o XML). Esta tecnología también es un desarrollo de la compañía Sun Microsystems.

4.5.1.9 JQUERY

jQuery [11] es un framework de JavaScript libre y de código abierto.
Se decidió utilizar el mismo, por sobre JavaScript, porque comparado con éste simplifica notoriamente la interacción con elementos HTML, el acceso a los elementos DOM (Document Object Model), el desarrollo de animaciones y agregar interacciones con AJAX en páginas web.

4.5.1.10 JSON

JavaScript Object Notation (JSON) [12] es un formato para el intercambio de datos, ligero y basado en un subconjunto de la sintaxis de JavaScript.
Tiene la ventaja de que se pueden incluir dentro de archivos JavaScript y acceder a su información sin la necesidad de incluir análisis adicional como por ejemplo ocurre con los lenguajes basados en XML (parser).
Además, el flujo de información extra (tags) que es necesario transmitir junto a los datos en JSON es significativamente menor al requerido en XML, lo cual le ha hecho ganar gran popularidad en aplicaciones web.

Por las ventajas mencionadas anteriormente y su simplicidad de uso es que se decidió utilizarlo para manejar datos (objetos o listas) en los archivos JavaScript.
En el proyecto, se utilizó la librería json-simple-1.1.1, la cual es una librería java para codificar/decodificar objetos JSON.

4.5.2 MYSQL

MySQL [27] es un sistema de gestión de bases de datos relacionales, que al ser software libre (bajo licencia GLP), permite su uso gratuitamente (también existe una versión comercial de MySQL); además, su diseño multihilo le permite manejar gran cantidad de datos de manera muy eficiente (velocidad alta y bajo consumo de recursos).
En la actualidad, es el gestor de bases de datos de código abierto más utilizado a nivel mundial.

Por otro lado, también se utiliza la herramienta MySQL WorkBench [10], la cual nos permite crear, diseñar, administrar y mantener bases de datos MySQL. De esta manera, nos es posible crear la base de datos a partir de un diagrama entidad-relación (diseñado previamente).

Las principales alternativas que se manejaron como gestores de bases de datos (todas de licencia gratuita) fueron PostgreSQL y MariaDB.

Si bien PostgreSQL es de los gestores de bases de datos (libres) más potentes, con gran robustez y escalabilidad, también es cierto que es bastante más lento que MySQL; por otra parte, MariaDB, al ser un fork de MySQL, no cuenta con las funcionalidades ni robustez que nos brinda MySQL.

Además, las herramientas de administración de bases de datos que provee MySQL (por ejemplo MySQL WorkBench) son bastante más completas que las presentadas por los otros dos mencionados anteriormente.

Finalmente, MySQL es por lejos la más popular y utilizada de las tres, lo que conlleva a una gran confiabilidad, mayor documentación y una comunidad muy activa.

En el proyecto se utiliza la versión 5.5 de MySQL y la 5.2 de MySQL WorkBench .

4.5.3 GOOGLE MAPS API

Para la implementación de la capa cliente del proyecto se decidió por utilizar un mapa de Google Maps [52] para desplegar los resultados visualmente, debido a la gran facilidad de incorporar uno de estos mapas a una aplicación web. Se implementaron prototipos en el comienzo del proceso de implementación para corroborar que con esta herramienta se podía lograr el objetivo visual que se buscaba.

Debido a la gran variedad de tutoriales en español para incorporar un mapa en una aplicación, se pudo comprobar que la herramienta satisface los requerimientos para poder mostrar la información que se genera en la capa lógica.

La principal funcionalidad que impulsó a utilizar la herramienta que brinda Google Maps, es que se puede obtener datos de distancias entre pares de puntos mediante una simple llamada al servicio web con los parámetros correspondientes. Al no contar con una base de datos con valores de distancias entre lugares geográficos de Uruguay, ésta es una buena solución para obtenerlos [58].

Además de la visualización del mapa base proporcionada por Google, los nodos sobre el mapa y las polilíneas resultantes de la distribución solución, Google Maps nos permite utilizar servicios WMS, brindando la posibilidad de por ejemplo agregar capas con otros datos georeferenciados sobre el mapa. El SINAE provee servicios WMS con la ubicación de los históricos de las inundaciones en Uruguay.

A continuación se presentan otras ventajas adicionales que alentaron a su uso para el

prototipo. Luego se presenta una serie de desventajas que evidencian que para la realización de una próxima versión del software debe evaluarse la posibilidad de contar con otra fuente de datos.

Ventajas:

- Google Maps es uno de los sistemas de localización más utilizados de internet. La posibilidad de mostrar fotos por satélite de cualquier parte del mundo es su característica más popular, pero también permite localizar direcciones y planificar rutas.
- Se puede utilizar en cualquier PC sin necesidad de instalar programas (se necesita un navegador) ni ocupar lugar en el disco duro.

Desventajas:

- Tiempo de respuesta de los servicios utilizados, por ejemplo, el que provee las distancias entre dos puntos, es muy grande y depende fuertemente de las características de la conexión a internet.
- Luego de muchas pruebas se encontraron varias localidades para las cuales, el servicio de distancias no funciona correctamente. En algunos casos se producen fallas en el servicio y no se puede obtener la respuesta deseada con el valor del tiempo transcurrido para trasladarse entre distintos puntos geográficos de Uruguay.
- Dependencia con el estado de servidores externos proporcionados por terceros. Imposibilidad de modificar los componentes implementados por terceros.
- Límite de uso, existe una limitante a la licencia libre que consiste en que se permiten realizar consultas de hasta 225 elementos cada 10 segundos; la cantidad de elementos a consultar en nuestro caso se calcula multiplicando la cantidad de orígenes por la cantidad de destinos (lo que resulta en una cantidad máxima de 15 nodos)

Estas desventajas condenan al uso de la herramienta e impulsan a que una futura versión del sistema utilice otra fuente de datos. La siguiente etapa de desarrollo debe consistir en la migración de la fuente de datos; desde el comienzo de la implementación, se intentó utilizar una arquitectura que, ante un cambio de esta magnitud, tenga el menor impacto sobre las restantes capas y o módulos del sistema.

Elementos utilizados de la interfaz:

- Map: Es la clase para mostrar un mapa, incluye las herramientas básicas de zoom.
- Marker: Es la clase utilizada para representar los nodos demandantes y los nodos proveedores.

- Polyline: Utilizada para representar las correspondencias entre los nodos proveedores y demandantes luego de calculada una distribución.
- El API de matriz de distancia de Google: Este servicio es utilizado para obtener los valores que corresponden al tiempo aproximado en segundos.
- API de rutas: Este servicio es utilizado para visualizar las rutas a utilizar para transportar los suministros desde los proveedores a los demandantes.

4.5.4 GLPK

Se necesita un módulo para resolver el modelo planteado. Para esto se utiliza un solver (programa que utiliza distintos algoritmos para intentar resolver problemas de optimización sujetos a restricciones), en esta investigación interesa el caso de restricciones lineales, con variables enteras y continuas (MIP, Mixed Integer Programming).

Para la resolución del problema de operación combinatoria del proyecto se decidió utilizar el solver GLPK [18]. Es un paquete escrito en lenguaje C que se utiliza para modelar y resolver problemas de optimización, de programación entera mixta:

- Pertenece a la comunidad de software libre.
- Ambos integrantes del equipo cuentan con experiencia en su uso, que es utilizado en otros cursos del Instituto de Computación de Facultad de Ingeniería.

Se logra implementar un prototipo al comienzo de la etapa de desarrollo para determinar la factibilidad de integración del módulo de GLPK con un proyecto Java EE [5].

4.5.5 ASSEMBLA WORKSPACE

Assembla Workspace es un conjunto de herramientas (disponibles online) que nos permiten organizar y administrar de manera eficiente proyectos de software. Se ofrecen licencias pagas, pero también existe la posibilidad de obtener una licencia gratuita (con capacidad de almacenamiento limitada, entre otros puntos).

Entre los principales servicios que nos brinda están los repositorios svn (utilizado en el proyecto) y Git, integrando herramientas que posibilitan la navegación y revisión de código; también ofrece la posibilidad de realizar asignación y seguimiento de tareas (por

ejemplo bugs o funcionalidades) a los usuarios asociados al proyecto. [16]

Existen muchas otras plataformas para almacenar proyectos de software, como por ejemplo Google Code [59] o SourceForge [42]. El problema que tiene Google Code es que no permite tener proyectos con código privado (por lo que el código puede ser visto y descargado por cualquier persona); por otra parte, si bien SourceForge es una buena alternativa, ambos integrantes del equipo ya conocíamos y habíamos trabajado en el pasado con Assembla Workspace, por lo que en función de las similares funcionalidades de ambos gestores de proyectos, terminamos inclinándonos finalmente por Assembla.

4.6 COMPONENTES DE LA SOLUCIÓN

4.6.1 ARQUITECTURA DEL SISTEMA

Para la implementación de la solución, se utilizó la plataforma Java EE [5] (específicamente JEE 6). Como ya mencionamos, una de las principales bondades que nos brinda esta plataforma es la de poder dividir el sistema en n-capas, según necesitemos, brindando modularidad al proyecto e independencia entre cada una de las capas.

Para el proyecto desarrollado, se optó por modularizar el proyecto en 4 capas: capa Cliente, capa Web, capa de Negocio y capa de Datos.

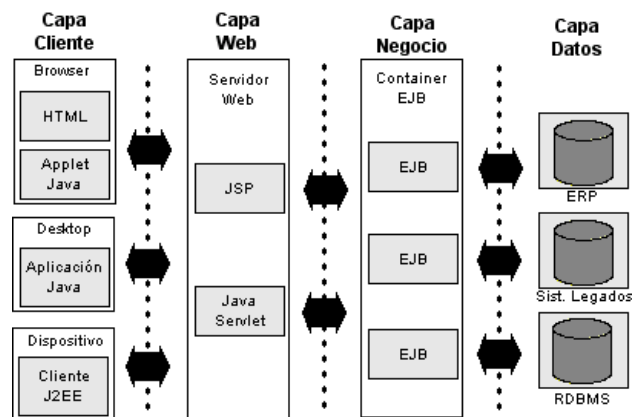


Figura 1 - Arquitectura en Capas [13]

A su vez, la interacción con Google Maps se produce en la Capa Cliente, realizándose pedidos (por ejemplo distancias o coordenadas de un punto geográfico) y obteniéndose las respuestas correspondientes del servidor de Google Maps.

La manera en la que se integra la aplicación con los servicios web de Google es similar a la que se representa en el siguiente diagrama:

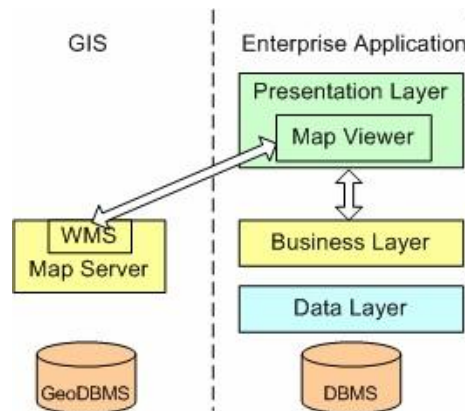


Figura 2 - Integración Aplicación-GIS [53]

4.6.1.1 CAPA CLIENTE

La capa Cliente, básicamente está compuesta por la interfaz gráfica del sistema y se encarga de interactuar con el usuario final (principalmente compuesta por código HTML, JavaScript y el uso de librerías de jQuery [11]). La interacción con la capa Web se produce principalmente mediante pedidos/respuestas utilizando el protocolo HTTP.

4.6.1.2 CAPA WEB

La capa Web es la encargada de manipular los pedidos realizados por el cliente, de esta manera, aquí es donde se encuentra la lógica de presentación (para mostrar al cliente) y también, en caso de ser necesario, se envían los datos recibidos por el cliente a la capa inferior para su procesamiento generando una respuesta adecuada al cliente. Esta capa básicamente está compuesta por componentes JSP [9] (Java Server Pages) y servlets.

4.6.1.3 CAPA DE NEGOCIO

La capa de Negocio es la responsable de procesar los datos obtenidos y devolver un resultado, de esta manera, aquí es donde se implementa toda la lógica de negocio del sistema, como ser el cálculo de la solución utilizando el GLPK o la heurística. Algunos de los componentes principales de esta capa son, entre otros, las entidades del sistema (mapeados a partir de la base de datos utilizando la API de JPA [8]), los EJB (Enterprise JavaBeans [7]) encargados de la comunicación con la capa de Datos, y los módulos asociados al cálculo de la distribución.

4.6.1.4 CAPA DE DATOS

La capa de Datos es donde residen los datos y se encarga de acceder a los mismos, en nuestro sistema, está conformado básicamente por la base de datos y por el gestor de base de datos JDBC [6] (Java Database Connectivity) encargado del almacenamiento y recuperación de datos de la base.

4.6.2 ESTRUCTURA Y FUNCIONAMIENTO DE LA IMPLEMENTACIÓN

La estructura se compone de un proyecto principal, llamado ProyectoLH, el cual contiene dos sub-proyectos asociados (ProyectoLH-ejb y ProyectoLH-war) y algunos archivos de configuración, como ser datos de conexión con la base (usuario y contraseña), pagina desde donde arranca la aplicación y servidor que se utiliza para correr la aplicación (GlassFish 3.1.2).

4.6.2.1 PROYECTOLH-EJB

El proyecto ProyectoLH-ejb básicamente es donde se realiza la lógica de negocio del sistema, siendo el responsable de calcular la distribución solución (capa de Negocio), y realizar las transacciones (capa de Negocio) y conexión (capa de Datos) con la base de datos.

Para el cálculo de la distribución solución, se obtienen los datos a partir de la base de datos, y se calcula la solución utilizando el modelo descrito en el punto 4.3. Para tal fin, se realizaron dos implementaciones del modelo, una utilizando el solver GLPK, que busca obtener la solución óptima, y otra utilizando una heurística [punto 4.4] que busca obtener una solución factible cuando el volumen de los datos a procesar lo amerite; de esta manera, el sistema calcula la solución (distribución) a partir del solver GLPK o la Heurística (según corresponda).

La manera en la que el sistema opera al momento de calcular la distribución es la siguiente:

- Si la cantidad de suministros existentes es suficiente como para abastecer toda la demanda se calcula la solución óptima utilizando el solver GLPK; en caso contrario directamente se calcula la solución utilizando la Heurística.
- Si se obtiene una solución óptima, se la pasa a la capa superior (capa Web) para mostrársela al cliente.
- En caso de que el cálculo de la solución utilizando el solver demore más de N segundos (actualmente se encuentra seteado en 2 segundos), o el solver no sea capaz de obtener la solución óptima, se calcula una buena solución

factible utilizando la Heurística. Luego, al igual que el caso anterior, se pasa la solución a la capa superior para que pueda ser mostrada al cliente.

Por otra parte, en este proyecto también se encuentran las entidades del sistema (mapeadas a partir de la base de datos utilizando JPA) y los Enterprise JavaBeans (EJB) los cuales se utilizan para poder realizar transacciones con la base de datos.

Finalmente, también se encuentra la unidad de persistencia (archivo XML) responsable de establecer la conexión con la base de datos utilizando la API Java Database Connectivity (JDBC) para MySQL.

4.6.2.2 PRIYECTOLH-WAR

El proyecto ProyectoLH-war básicamente es donde se realiza la interacción con el cliente, presentándole las distintas interfaces existentes (capa Cliente) y trabajando con los datos ingresados por el mismo (capa Web).

Para procesar y trabajar con los datos ingresados por el cliente, se utilizan servlets (Java Servlet) y paginas JSP (JavaServer Pages) [13].

La función de los Servlets en el proyecto fundamentalmente es la de recibir los datos de la capa superior (capa Cliente), procesarlos y enviar la respuesta adecuada de nuevo a la capa superior. De esta manera, cuando el Servlet recibe (de la capa superior) la acción que quiere realizar el usuario (por ejemplo calcular la distribución, mostrar los atributos de un nodo, etc.), envía dicha información a la capa inferior (capa de Negocio) para su procesamiento. Finalmente se obtiene una respuesta de dicha capa y se invoca a la página JSP adecuada (pasándole los datos correspondientes) para mostrarle la respuesta adecuada al usuario.

Por otra parte, las paginas JSP son las responsables de generar el código HTML que será mostrado al usuario. De esta manera generan el HTML correspondiente a partir de los datos pasados por el Servlet invocante o datos proporcionados por la sesión.

Para presentarle el sistema al usuario básicamente se cuenta con código HTML (generado por los JSP), los archivos de estilo (CSS), archivos de código JavaScript y utilidades de jQuery.

El HTML y CSS tienen la simple misión de mostrarle al usuario la página con el estilo y formato correcto.

Por otro lado, el jQuery cumple con las funciones de validación y verificado de los campos que ingresa un usuario (antes de enviar los datos al servidor), animaciones en el HTML (por ejemplo desplegar menús, etc.) y diversas acciones asociadas a la lógica de la

interfaz y también en algunos casos envíos (post) de datos al servidor.

En la capa cliente se implementa una compleja lógica con el objetivo de minimizar la cantidad de llamadas al servidor y por consecuencia optimizar el tiempo de ejecución.

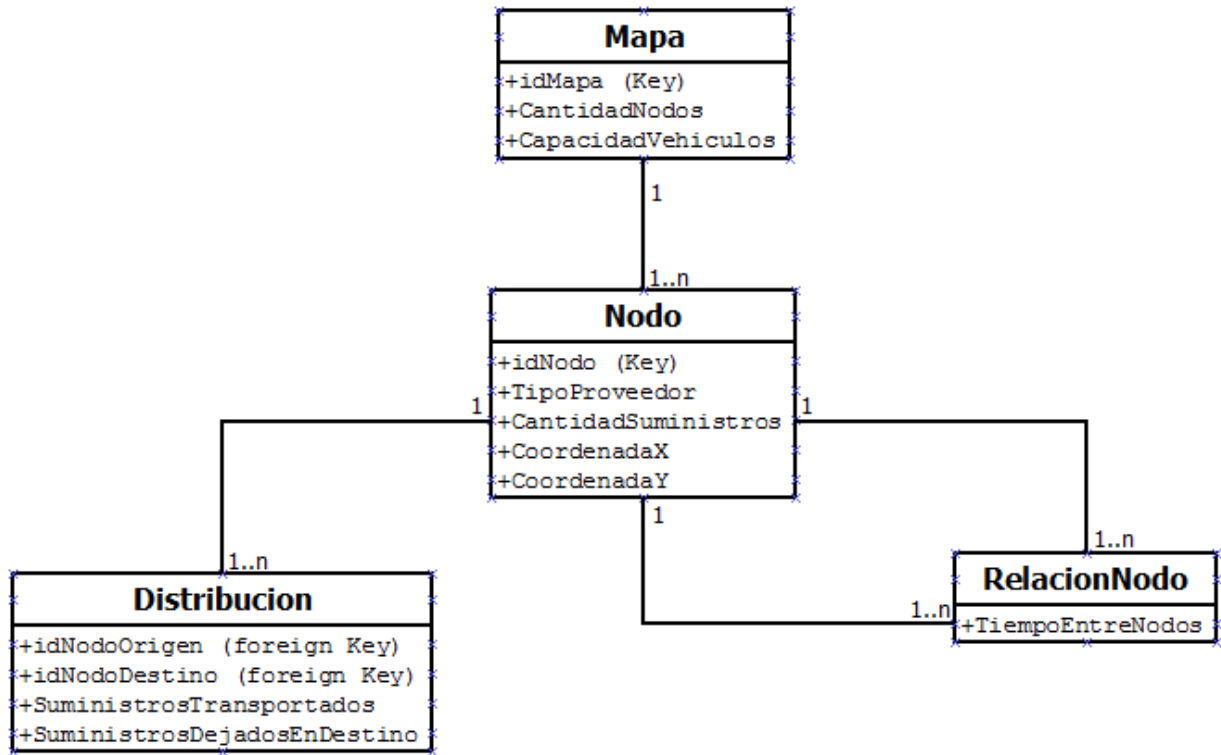
El funcionamiento básico es de la siguiente manera: se mantienen dos listas en memoria, una para los nodos de tipo proveedor y otra para los de tipo demandante. A la hora de calcular solución, si no se conoce aún la distancia entre los nodos, se realiza un pedido al servicio de matriz de distancias de Google Maps. Cuando se recibe la respuesta, se manda al servidor web para comenzar el proceso de encontrar la solución. Cuando llega esta respuesta, se despliega en el mapa las polilíneas de los nodos representando: si existe polilínea entre proveedor A y demandante B quiere decir que se debe visitar A y luego B. Puede verse (mediante un menú) el detalle de cada trayecto entre 2 nodos identificando el orden de visita de los nodos y la cantidad de suministros transportados en cada tramo, así como también la cantidad de suministros que se dejan en cada demandante.

El API JavaScript de Google Maps permite la visualización del mapa de Google, así como los nodos sobre el mapa y las líneas que representan la distribución solución; a su vez, también permite utilizar servicios WMS, brindando la posibilidad de por ejemplo agregar capas con datos georeferenciados sobre el mapa (como los datos históricos proporcionados por el SINAIE).

4.6.3 MODELO DE LA BASE DE DATOS

Como se mencionó anteriormente, para la realización del proyecto se eligió el gestor de base de datos MySQL utilizando la herramienta MySQL Workbench [10] para el diseño, creación y administración de la base.

A continuación mostraremos el diseño de la base y explicaremos cada uno de sus componentes y relaciones.



Como se puede observar en el diagrama, las entidades de la base son las siguientes:

- Mapa
- Nodo
- Distribución
- RelacionNodo

4.6.3.1 MAPA

La entidad Mapa contiene como clave primaria el atributo `idMapa`, el cual lo identifica dentro del sistema (nombre con el cual se guarda en el sistema).

El atributo `CantidadNodos`, como su nombre lo indica, corresponde a la cantidad de nodos que tiene asociados el mapa; este valor se va actualizando conforme se agreguen o eliminen nodos del mismo.

A su vez, el atributo `CapacidadVehiculos` define la capacidad de los vehículos que se utilizan en el mapa.

4.6.3.2 NODO

La relación existente entre las entidades Nodo y Mapa es 1 – 1 a n, lo cual significa que un Mapa puede tener entre 1 y n Nodos asociados, mientras que un Nodo está asociado únicamente a un Mapa.

La entidad Nodo contiene como clave primaria el atributo idNodo, el cual, junto con el identificador del mapa asociado (idMapa), lo identifica dentro del sistema (esto sucede en parte por la relación existente entre las dos entidades).

Los atributos CoordenadaX y CoordenadaY, como sus nombres lo indican, son las coordenadas “x” e “y” (obtenidas de Google Maps) que tiene el Nodo.

Además, también posee los atributos TipoProveedor y CantidadSuministros. El primero de ellos sirve para identificar si un nodo es del tipo Proveedor o Demandante; y el segundo sirve para indicar, en caso que el Nodo sea del tipo Proveedor, la cantidad de suministros que se encuentran en el Nodo (suministros a distribuir), y si el nodo es del tipo Demandante indica la cantidad de suministros demandados por el Nodo. Vale aclarar que, cuando se calcula una distribución para un determinado Mapa, los valores del atributo CantidadSuministros de los Nodos asociados a dicho Mapa, se van a modificar de manera que indicarán la cantidad de suministros que no se distribuyeron y quedaron en el nodo (en caso de que el nodo sea del tipo Proveedor), o indicarán la cantidad de demanda insatisfecha en el nodo (en caso de que el nodo sea del tipo Demandante).

4.6.3.3 RELACIONNODO

La entidad RelacionNodo indica la relación existente entre dos pares de Nodos, de manera que si existe un camino entre dos pares de nodos (se toma en cuenta el sentido), este se representará a través de esta entidad.

De esta manera, se puede observar que existe una relación doble 1 – 1..n entre las entidades Nodo y RelacionNodo, indicando así la relación existente entre dos pares cualesquiera de Nodos.

El único atributo que tiene la entidad RelacionNodo es TiempoEntreNodos, el cual muestra el tiempo necesario para llegar desde un nodo hasta otro.

Cabe mencionar que el valor con el cual se “setea” el atributo TiempoEntreNodos es obtenido a través de la API de Google Maps.

4.6.3.4 DISTRIBUCIÓN

Las entidades Distribución se utilizan para guardar la distribución calculada.

Cada entidad Distribución va asociada a un solo Nodo, el cual debe de ser del tipo Proveedor, y a la vez cada nodo del tipo Proveedor puede tener asociados 1..n entidades Distribución.

En esta entidad, el atributo SuministrosTransportados indican la cantidad de suministros que se transportan entre dos pares de Nodos, mientras que el atributo SuministrosDejadosEnDestino indica la cantidad de estos suministros que son dejados en el Nodo destino.

Por otra parte, los atributos idNodoOrigen e idNodoDestino indican los nodos origen y destino entre los cuales se transportan los suministros. Vale aclarar que para que exista una entidad Distribucion, debe existir una entidad RelacionNodo entre los nodos identificados por los atributos idNodoOrigen e idNodoDestino de la entidad Distribución. De esta manera, cada Nodo Proveedor tendrá su propia “lista” de entidades Distribución, la cual indicará al Proveedor los nodos que debe ir visitando (en orden) y la cantidad de suministros que debe dejar en cada uno.

4.7 PROCESO DE DESARROLLO

Para la realización del software se utilizó un proceso de desarrollo en cascada compuesto por las etapas de Análisis, Investigación, Diseño, Implementación, Integración y Pruebas.

4.7.1 ANÁLISIS DEL PROBLEMA

En esta primera etapa se definió, en conjunto con los tutores, el problema que se intentaba resolver.

Luego, se definió el alcance del proyecto, especificando los requerimientos y funcionalidades principales del sistema a desarrollar.

Finalmente, se definieron las etapas a seguir en el proceso de desarrollo y se hizo un estimativo bruto del tiempo necesario para la realización de cada etapa.

4.7.2 INVESTIGACIÓN

Una vez definido el problema a resolver y las principales funcionalidades del mismo, se procedió a realizar una investigación sobre qué tecnologías utilizar para poder llevar a cabo la realización del software.

No existiendo restricciones sobre el uso de tecnologías específicas se realizó una investigación sobre algunos aspectos importantes a la hora de realizar el desarrollo de software, como ser lenguaje de programación, plataforma, base de datos, GIS, etc., realizando múltiples pruebas y prototipos al inicio de la etapa de implementación para comprobar que con las herramientas elegidas realmente se podía resolver el problema.

4.7.3 DISEÑO DE LA SOLUCIÓN

En esta etapa se definió la arquitectura del sistema, optándose por modularizar el proyecto en cuatro capas: capa Cliente, capa Web, capa de Negocio y capa de Datos; a su vez, se definen las funcionalidades y roles de cada una de estas capas dentro del sistema.

Esto nos brinda la capacidad de abordar un problema que en un principio era complejo y de gran tamaño en problemas más pequeños y sencillos, a la vez que nos permite paralelizar trabajo y una mejor y más sencilla reutilización de código.

También se define el modelo de la base de datos, especificando las distintas entidades (junto con sus atributos) así como las relaciones entre cada una de ellas.

4.7.4 IMPLEMENTACIÓN

En la etapa de Implementación se realiza el desarrollo de las capas mencionadas anteriormente, paralelizándose en algunos casos; por ejemplo algunas funcionalidades de las capas Cliente y Web se realizaron en paralelo al desarrollo de las capas de Negocio y Datos.

4.7.5 INTEGRACIÓN

En esta etapa se procedió a integrar las distintas capas que forman parte del sistema. En primer lugar, se integraron las capas de Negocio con la de Datos y la capa Cliente con la Web. La manera por la cual se procedió así fue porque dichas capas se encuentran más ligadas que las restantes; luego, en una etapa final, se procedió a integrar la capa Web con la de Negocio, logrando así la integración del sistema completo.

4.7.6 PRUEBAS

En esta etapa se definen y ejecutan un conjunto de pruebas a realizarse sobre el sistema para comprobar el correcto funcionamiento del mismo.

De esta manera, se especificaron un conjunto de escenarios, variando la cantidad de nodos y la distribución de los mismos en el mapa, de manera de poder probar tanto la calidad de la distribución solución obtenida así como también el tiempo necesario para calcular la misma.

4.7.7 IDENTIFICACIÓN Y PLANIFICACIÓN DE TAREAS PARA EL DESARROLLO

En esta sección se especifica la identificación inicial de tareas a realizar en la etapa de implementación del prototipo, se presentan en el orden cronológico en el que fueron atacadas; cabe destacar que varias de éstas tareas se realizaron en paralelo:

- 1- Investigación de herramientas:
 - a. Prototipo de integración de un proyecto del IDE Netbeans del tipo JavaEE con la herramienta GLPK.
 - b. Prototipo de integración de interfaz grafica implementada con las herramientas de Google Maps:
 - i. Pruebas del servicio de ruteo.
 - ii. Pruebas del servicio de la matriz de distancias.
 - iii. Pruebas manipulando Markers y Polilineas.
- 2- Implementación de la lógica, capa de negocio:
 - a. Módulo de resolución utilizando GLPK.
 - b. Módulo de resolución utilizando la heurística.
- 3- Implementación de la capa de datos.
- 4- Implementación de la capa web.
- 5- Implementación de la interfaz gráfica
- 6- Integración de los distintos componentes:
 - a. Integración de capa de datos con la de negocio.
 - b. Integración de capa de negocio con la web.
 - c. Integración de la capa web con la interfaz gráfica.
- 7- Pruebas

- a. Pruebas de Módulo de resolución utilizando GLPK.
- b. Pruebas de Módulo de resolución utilizando la heurística.
- c. Pruebas del sistema completo con la Interfaz de usuario.

5 PLAN DE PRUEBAS

5.1 METODOLOGÍA

Las pruebas a realizarse deben verificar y validar el correcto funcionamiento del sistema, considerando el alcance del proyecto [punto 4.1] así como también los requerimientos funcionales y no funcionales definidos [punto 4.2].

Como ya se ha mencionado, uno de los puntos fundamentales del sistema es poder brindarle al usuario una distribución solución de manera rápida, y a la vez de calidad; por lo tanto, resulta crucial para las pruebas especificar el hardware sobre el que se correrán las mismas, pues los tiempos de procesamiento pueden verse afectados en función del mismo, principalmente interesa el procesador y memoria RAM.

La computadora sobre la que se ejecutarán las pruebas consta de un procesador Intel Core i5-2410M (2.5GHz) y 6 GB (DDR3) de memoria RAM.

En función de la importancia de los tiempos de resolución y de las distintas formas en las que el sistema puede calcular la distribución solución, se plantean tres conjuntos de pruebas, cada una con un conjunto de escenarios variados.

De esta manera, los tres conjuntos de pruebas a realizarse serán:

- Pruebas del Modelo utilizando el solver GLPK
- Pruebas de la Heurística
- Pruebas del Sistema completo

Los dos primeros conjuntos de pruebas estarán destinados a verificar los tiempos de resolución necesarios para calcular la distribución solución, así como también la calidad de la misma. Esto último se realiza básicamente verificando si se obtiene una correcta distribución entre los nodos existentes (por ejemplo observando si se atienden a los nodos más cercanos) así como también si se corresponde con una distribución lógica y no quedan suministros sin repartir cuando aún exista demanda.

Para estos dos conjuntos de pruebas se plantean distintos escenarios que contemplen tanto la cantidad de nodos, la cantidad de cada tipo de nodo (Proveedores y Demandantes), la cantidad de demanda y suministros, y la topología en la que se encuentran distribuidos cada uno de los nodos demandantes y proveedores.

Finalmente se realizan pruebas sobre el sistema en general, verificando tanto la performance del sistema (tiempos para calcular y mostrar la distribución), así como

también los requerimientos funcionales y no funcionales especificados [punto 4.2].

5.2 PRUEBAS CORRIDAS Y RESULTADOS OBTENIDOS

En esta sección, se plantean las pruebas corridas sobre el sistema así como también los resultados obtenidos en cada una de ellas.

Las pruebas que se efectuaron, como se especifica en el plan de pruebas, pueden dividirse en tres secciones:

- Pruebas del modelo utilizando el solver GLPK
- Pruebas de la Heurística
- Pruebas del sistema completo

5.2.1 PRUEBAS DEL MODELO UTILIZANDO SOLVER GLPK

Estas pruebas buscan medir la performance del sistema para calcular la distribución utilizando el GLPK. Específicamente, involucran la resolución con el GLPK y las interacciones necesarias con la base de datos (como consultas o guardar la distribución obtenida).

Es necesario recordar que al resolver el modelo utilizando el GLPK, lo que se obtendrá será la distribución solución óptima, y considerando que el problema a resolver es del tipo NP-Hard, es de esperar que los tiempos de resolución a medida que se aumente la cantidad de nodos, crezca exponencialmente. A su vez, también es importante mencionar que el modelo solamente encontrará una solución cuando la cantidad de suministros de los nodos demandantes alcancen para satisfacer la demanda existente, pues de esa manera está diseñado el modelo (en otro caso se utiliza la heurística).

A continuación se plantea una tabla con las pruebas realizadas junto con los resultados obtenidos.

La estructura de la tabla es la siguiente:

- La primer columna indica el número de prueba
- La segunda columna indica el número de nodos del tipo demandante en la prueba
- La tercer columna indica el número de nodos del tipo proveedor en la prueba
- La cuarta columna indica la cantidad de suministros demandada por cada uno de los nodos demandantes, y la cantidad de suministros totales demandados en la prueba (separados por un guion "-")
- La quinta columna indica la cantidad de suministros que posee cada uno de los nodos proveedores, y la cantidad de suministros totales disponibles en la

prueba (separados por un guion "-")

- La sexta columna indica el escenario (topología y suministros) en el que se encuentran distribuidos los nodos (se especifican los tiempos entre cada par de nodos)
- La séptima columna indica el tiempo necesario (en segundos) para hallar la distribución solución (tiempo que le lleva al solver calcular la solución)
- La octava columna indica si la distribución obtenida se corresponde a la solución esperada

Nº prueba	Nº Demandantes	Nº Proveedores	Suministros por Demandante - Suministros Totales	Suministros por Proveedor - Suministros Totales	Escenario	Tiempo (seg)	Distribución
1	6	2	2 - 12	12 - 24	Uniforme	21.7	OK
2	6	2	2 - 12	6 - 12	Uniforme	18.5	OK
3	2	6	5 - 10	2 - 12	Uniforme	35.2	OK
4	7	2	2 - 14	14 - 28	Uniforme	78.4	OK
5	7	2	2 - 14	7 - 14	Uniforme	51.9	OK
6	8	2	2 - 16	16 - 32	Uniforme	---	---
7	8	2	2 - 16	8 - 16	Uniforme	---	---
8	6	2	2 - 12	7 - 14	Zona autosuficiente	15.4	OK
9	6	2	2 - 12	-- - 14	Zona no autosuficiente	17.6	OK
10	2	2	3 - 6	4 - 8	Uniforme	1.3	OK
11	3	1	2 - 6	6 - 6	Uniforme	0.8	OK

Aclaraciones de Escenarios:

- Uniforme: se refiere a una topología de red donde el tiempo entre todo par de nodos en la prueba es el mismo. Es importante mencionar que el modelo no verifica que se cumpla la desigualdad triangular, es más, cuando trabajamos con tiempos no necesariamente se debe cumplir la misma.
- Zona autosuficiente: se refiere a una topología de red donde cada nodo proveedor tendrá un conjunto de nodos demandantes cercanos (estando los restantes nodos demandantes alejados con este proveedor), de esta manera se forman zonas donde cada proveedor tiene un conjunto de nodos demandantes cercanos. A su vez, en este escenario la cantidad de suministros que tiene el nodo proveedor es suficiente para satisfacer la demanda de sus nodos demandantes más cercanos (zona).
- Zona no autosuficiente: se maneja la misma topología que en el caso anterior (zonas donde cada proveedor tiene un grupo de demandantes cercanos), pero en este caso algunos de los proveedores no tiene suficientes suministros

como para abastecer la demanda de su “zona”, mientras que otros cuentan con suministros suficientes como para abastecer su zona y aún más.

A partir de los resultados obtenidos en las pruebas 1 y 2, se observa que al variar solamente la cantidad de suministros, se nota una diferencia importante en los tiempos de resolución, lo cual resulta lógico pues al ser mayor la diferencia entre la demanda y suministros existentes las posibilidades de realizar la distribución para cada proveedor aumentan (cada proveedor puede abastecer más demandantes), teniendo en cuenta que se busca la solución óptima (minimizar el tiempo).

A su vez, a partir de la prueba 3, también se puede observar que si se aumenta la cantidad de proveedores, también aumenta el tiempo de ejecución. Esto se debe a que cada proveedor posee vehículos con los cuales realiza la distribución, y cuanto más vehículos haya, mayor serán las cantidades de posibles rutas solución (una ruta por cada vehículo), por lo que se necesitará mayor cantidad de operaciones (más rutas que calcular y comparar) para hallar la solución óptima, lo que resulta en mayor tiempo de computo.

Como se mencionó anteriormente, a medida que aumenta la cantidad de nodos también lo hace el tiempo necesario para calcular la solución. Esto se puede observar en las pruebas 4, 5, 6 y 7, en donde en las pruebas con 9 nodos aumenta sensiblemente el tiempo comparado con las pruebas con 8 nodos (pruebas 1, 2 y 3).

Finalmente, cuando se realizan las pruebas con 10 nodos no se llega a obtener una solución; más específicamente lo que sucede es que la ejecución del solver no termina pasados los 2 minutos y comienza a consumir muchos recursos de la máquina (memoria y procesador) debiendo terminar con el proceso. Vale aclarar que para el caso de 10 nodos, en cada iteración cada vehículo en cada nodo puede elegir entre 9 posibles caminos (uno por cada nodo vecino), ya que se permite volver al mismo nodo del que se vino (por ejemplo si el tiempo asociado a esa arista así lo amerita).

La prueba 8 posee 2 proveedores y cada uno de ellos tiene 3 demandantes cercanos (zona del proveedor) y 3 lejanos. El resultado obtenido es que cada proveedor distribuye a sus demandantes más cercanos (zona), por lo que se logra una correcta zonificación y distribución.

Vale aclarar que cuando se hace alusión a los términos “cercano” y “lejano” no se refiere a distancia sino a tiempo necesario para llegar al nodo mencionado.

En la prueba 9 se tiene la misma topología que la mencionada anteriormente, pero en este caso uno de los proveedores (A) posee 10 suministros (suficientes para abastecer su zona – 6 suministros), mientras que el otro proveedor (B) solamente tiene 4 suministros (insuficientes para abastecer su zona – 6 suministros).

De esta manera, la solución obtenida comprende al proveedor A atendiendo la demanda de su zona y un demandante de la zona del proveedor B, por su parte B atiende la demanda de dos de los demandantes de su zona, por lo que se logra satisfacer toda la

demanda del sistema.

Finalmente, considerando que uno de los requerimientos del sistema es brindar una rápida distribución solución, el sistema intentará calcular la solución utilizando el solver GLPK, y en caso que el mismo demore más de 2 segundos se termina ese proceso y se calculará la solución utilizando la heurística; este tiempo puede ser. Con 4 nodos los tiempos de resolución que se obtienen utilizando el GLPK son inferiores o rondan los 2 segundos (como se puede observar en las pruebas 10 y 11).

5.2.2 PRUEBAS HEURÍSTICA

En estas pruebas se busca medir la performance del sistema para calcular la distribución solución utilizando la Heurística, considerándose tanto el tiempo de computo asociado a la heurística en sí como también las interacciones necesarias con la base de datos (como consultas o guardar la distribución obtenida).

La heurística desarrollada busca reducir considerablemente los tiempos de resolución en comparación con los del GLPK a la vez de obtener una distribución solución de calidad. A continuación se plantea una tabla con las pruebas realizadas junto con los resultados obtenidos (la estructura de la tabla es la misma que la que se mostró en las pruebas con GLPK [punto 5.2.1]).

Nº prueba	Nº Demandantes	Nº Proveedores	Suministros por Demandante - Suministros Totales	Suministros por Proveedor - Suministros Totales	Escenario	Tiempo (seg)	Distribución
1	200	50	2 - 400	5 - 250	Uniforme	10.9	OK
2	50	200	10 - 500	2 - 400	Uniforme	13.8	OK
3	50	5	5 - 250	50 - 250	Zona autosuficiente	6.3	OK
4	50	5	5 - 250	-- - 190	Zona no autosuficiente	6.8	OK

Se aclara que los escenarios utilizados son los mismos que los presentados en las pruebas con el solver [punto 5.2.1].

A partir de los resultados obtenidos en las pruebas 1 y 2 se puede observar que existe una diferencia de tiempo cuando se varía la cantidad de nodos proveedores. Esto se debe a que a medida que aumenta la cantidad de nodos proveedores también aumenta la cantidad de vehículos y por ende aumenta la cantidad de rutas a considerar (una por

cada vehículo), todo esto lleva a que se deba realizar mayor cantidad de operaciones para encontrar una distribución solución, lo que resulta en mayor tiempo de computo. Es importante mencionar que para estos casos (250 nodos) la cantidad de entradas en una de las tablas de la base de datos (tabla que indica el tiempo necesario para recorrer cada arista del grafo, refiriéndonos a arista como el camino que une dos nodos vecinos) asciende a 62.250, teniendo un tamaño aproximado de 5 Mb.

La prueba 3 posee un escenario de “zona autosuficiente”, donde cada uno de los proveedores tiene 10 demandantes cercanos (zona del proveedor) y 40 lejanos. El resultado obtenido es el esperado, pues se obtiene una distribución donde cada proveedor se encarga de abastecer la demanda de su zona.

Vale aclarar, que al igual que en el caso de las pruebas con el solver, cuando se hace alusión a los términos “cercano” y “lejano” no se refiere a distancia sino a tiempo necesario para llegar al nodo mencionado.

En la prueba 4, al tenerse un escenario de “zona no autosuficiente”, se tiene la misma topología que en la prueba anterior, pero en materia de asignación de suministros se tiene el siguiente escenario:

- 50 demandantes, cada uno con una demanda de 5 suministros
- 3 proveedores (A, B, C) con 60 suministros cada uno
- 1 proveedor (D) con 10 suministros
- 1 proveedor (E) sin suministros

La distribución obtenida es la esperada, pues los proveedores A, B y C atienden toda la demanda de sus respectivas zonas, a la vez de satisfacer la demanda de algunos nodos demandantes de las zonas correspondientes a los proveedores D y E. Por su parte, el proveedor D atiende la demanda de dos de los demandantes de su zona (solamente tiene esa cantidad de suministros). Como resultado, no se logra satisfacer toda la demanda existente en la prueba, y los proveedores quedan sin ningún suministro.

Comparando los resultados obtenidos aplicando la Heurística con los conseguidos utilizando el solver GLPK claramente se puede observar que los tiempos de resolución difieren muchísimo; pero también es de interés comparar ambos métodos en función del costo asociado a sus soluciones (y no tanto del tiempo utilizado).

Para tal fin, se plantea el siguiente cuadro donde se utilizan 9 nodos (como ya se mencionó con 10 nodos el solver consume demasiados recursos de máquina y no llega a obtener la solución).

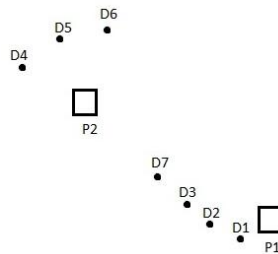
Nº prueba	Nº Demandantes	Nº Proveedores	Suministros por Demandante - Suministros Totales	Suministros por Proveedor - Suministros Totales	Método	Tiempo (seg)	Costo (seg)
1	7	2	20 - 140	80 - 160	GLPK	62.9	7
2	7	2	20 - 140	70 - 140	Heurística	1.0	8

Cada uno de los nodos proveedores en las pruebas tiene un vehículo, el cual tiene capacidad de 80.

El escenario con el que se corrieron las pruebas corresponde al de la siguiente figura, donde, para simplificar el problema, las distancias (tiempos) entre los nodos P2, D4, D5 y D6 es de 1 minuto (para todo par de nodos); lo mismo ocurre con las distancias entre los nodos P1, D1, D2 y D3.

La distancia entre el par (P2, D7) es de 4 minutos y (D6, D7) es de 2 minutos, mientras que entre el par (P1, D7) es de 3 minutos y entre el par (D3, D7) es de 1 minuto.

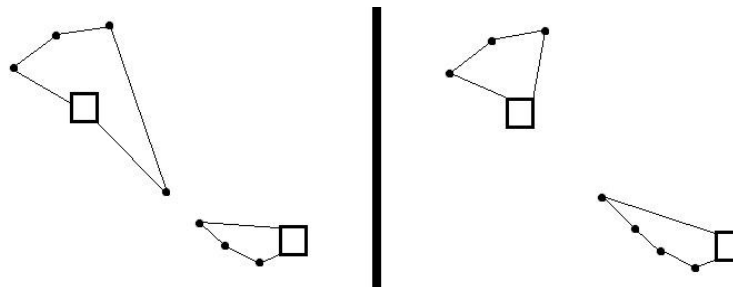
El resto de las distancias que no se especifican se suponen “infinitas” a efectos prácticos.



El dibujo busca ser simplemente un ejemplo ilustrativo del posible escenario, no reflejando de manera fiel los datos mencionados anteriormente.

Es importante mencionar que se buscó específicamente un escenario donde el solver arrojara un mejor resultado que la heurística (en términos de costo asociado a la solución), pues si se simplifica demasiado el escenario ambos métodos arrojarían resultados de igual coste. Se recuerda que el costo tanto de la heurística como del modelo (resuelto con el GLPK) corresponde a la suma de los tiempos de todos los caminos que formen la solución, sin considerar los caminos de vuelta desde los últimos clientes visitados hasta los proveedores.

Finalmente, a continuación se plantean los resultados obtenidos por la heurística (derecha) y el solver (izquierda), teniendo un costo asociado de 7 y 8 minutos respectivamente.



El motivo por el cual la heurística en este caso decide que el demandante D7 debe ser atendido por el proveedor P2 es porque la distancia de (P2, D7) es menor a (P1, D7), por lo que al momento de crear la zonificación en cada nodo se decide que D7 quede asociado a P2.

5.2.3 PRUEBAS DEL SISTEMA

Las pruebas a desarrollarse aquí se enfocan en el correcto cumplimiento de los requerimientos funcionales y no funcionales, haciendo especial hincapié en la performance del sistema.

Respecto a los requerimientos funcionales se logran cumplir con todos los puntos especificados en el punto 4.2.1, existiendo todas las funcionalidades descritas en el mismo y funcionando correctamente.

Uno de los casos de uso que se utilizó para verificar estos requerimientos se presenta a continuación:

- Se ingresa al sistema (con un navegador especificando la url correspondiente) y se crea un nuevo mapa con nombre “mapaPrueba” y se especifica la capacidad de los vehículos asociados a dicho mapa en 200 unidades.
- Se edita la capacidad de los vehículos a 1500 unidades y se guarda el mapa.
- Se agregan dos nodos proveedores (P1 y P2 con suministros 1400 y 300 respectivamente) y tres demandantes (D1, D2 y D3 con demandas 500, 600 y 200 respectivamente).
- Se calcula la distribución solución y en el mapa se muestran las rutas a utilizarse en la misma (polilíneas entre pares de nodos). A su vez se actualizan las cantidades de suministros y demandas en el mapa en función de la distribución calculada.
- Se selecciona mostrar la distribución y se muestran las rutas (calles específicas en google maps) de cada uno de los vehículos asociado a cada nodo proveedor, a su vez se especifica la carga de cada vehículo y la cantidad de suministros que quedan en cada uno de los nodos demandantes visitados.
- Se borran los nodos P2 y D3; luego se actualiza el nodo P1 con 500 suministros, y D1 y D2 con 100 y 300 unidades de demanda respectivamente, posteriormente se guarda el mapa.
- Se sale del sistema y se vuelve a ingresar (por ejemplo cerrando y volviendo a abrir el navegador). Luego se carga el mapa “mapaPrueba” verificando que los nodos, sus demandas y suministros, y la capacidad de los vehículos se encuentre como lo habíamos guardado.
- Se selecciona alguna de las capas de datos (servicios WMS) que proporciona el sistema y se verifica que se visualice correctamente.
- Se calcula nuevamente la distribución solución y la misma aparece en el mapa.

En relación a los requerimientos no funcionales,

- Se verifica que el sistema se visualiza (css y html) y funciona (jQuery y

JavaScript) correctamente en los navegadores Mozilla Firefox y Google Chrom.

- Se verifica que la distribución calculada siempre busca minimizar los tiempos necesarios para realizar las entregas, ya que tanto el modelo como la heurística buscan minimizar dicho tiempo.
- Se verifica la utilización de software libre para la realización del proyecto [punto 4.5].
- Se logra una interfaz clara y limpia, brindando al usuario final una clara lectura e interpretación de la distribución planteada, por lo tanto se verifica una correcta presentación de la distribución solución.

Es importante mencionar que la API de Google Maps, en su versión de licencia gratuita, permite consultas de hasta 225 elementos cada 10 segundos; la cantidad de elementos a consultar en nuestro caso se calcula multiplicando la cantidad de orígenes por la cantidad de destinos (lo que resulta en una cantidad máxima de 15 nodos); en la versión paga, sin embargo, no existen tales limitantes.

Finalmente, para verificar los tiempos necesarios para calcular y presentar la distribución solución, se plantea una tabla con distintos escenarios (considerando las limitaciones de la API mencionadas anteriormente) para probar la performance del sistema:

Nº prueba	Nº Demandantes	Nº Proveedores	Suministros por Demandante - Suministros Totales	Suministros por Proveedor - Suministros Totales	Tiempo (seg)
1	8	2	1500 - 12000	2500 - 5000	0.8
2	2	8	1500 - 3000	2500 - 20000	1.0
3	10	5	1500 - 15000	2500 - 12500	1.7
4	5	10	1500 - 7500	2500 - 25000	2.2

Los tiempos mostrados en esta tabla son bastantes superiores a los que se pueden obtener utilizando el solver GLPK [punto 5.2.1] o la heurística [punto 5.2.2], esto se debe a que en éste caso se envían y reciben datos de Google Maps (en los otros dos casos se trabaja localmente), y además la interfaz cuenta con gran cantidad de lógica (en archivos JavaScript) donde se realizan gran cantidad de cálculos y se llevan registros (listas) con información de los nodos existentes (en las pruebas anteriores no se trabajaba con la interfaz sino que los resultados se visualizaban a través de consultas directas con la base de datos); todo esto aumenta de manera significativa los tiempos necesarios para presentarle la solución al usuario final.

Cabe mencionar que en las pruebas se ha podido detectar que cuando se obtiene una ruta entre dos nodos (ruta asociada a una de las polilíneas correspondientes a la distribución), cuando mayor sea la distancia entre estos dos nodos, mayor será el tiempo que demore el sistema en mostrar esta ruta. Esto se debe a que cuanto mayor es la

distancia, mayor será el tiempo que le tome a Google Maps calcular y enviarnos la ruta solución, además de que la cantidad de datos a enviarse será mayor.

No se realizaron pruebas de carga ni pruebas de stress de la aplicación web porque el sistema no está pensado para que sea utilizado por una cantidad grande de usuarios, es un sistema de apoyo a la toma de decisiones operativas (operado en un principio por un conjunto no muy grande de usuarios -autorizados-).

Finalmente, creemos que se han logrado tiempos de resolución bajos y acordes a las expectativas planteadas, así como también una interfaz bastante intuitiva y que permite mostrarle al usuario de forma clara la distribución obtenida.

6 CONCLUSIONES

Las operaciones de logística humanitaria son altamente complejas, ya que se requiere responder de inmediato a situaciones no predecibles. Muchas veces no se conoce de antemano qué se demanda ni cuánto, esto dificulta las tareas relacionadas con la distribución de ayuda.

La aplicación de conocimientos, habilidades, la movilización de personas y recursos, influye positivamente en la rápida atención de la población afectada. La utilización de modelos y algoritmos de optimización, puede ayudar a que las operaciones logísticas humanitarias sean realizadas eficiente y efectivamente.

Comprender las diferencias y similitudes entre logística humanitaria y logística comercial es importante porque existe una amplia gama de modelos y algoritmos avanzados en la logística clásica que se pueden analizar, reutilizar y extender para mejorar la operativa en un contexto humanitario.

En este proyecto, nuestro primer objetivo fue realizar un relevamiento sobre Logística Humanitaria y en segunda instancia, desarrollar un modelo de distribución de suministros para optimizar el tiempo de respuesta en una situación de emergencia. En base a los modelos y algoritmos relevados, se desarrolló un nuevo modelo matemático simple que resulta de la aplicación de los conocimientos adquiridos durante la investigación. La implementación del modelo y las habilidades como desarrolladores de software se reflejan en el prototipo implementado.

El prototipo se compone de una aplicación web con una interfaz amigable que presenta, al momento de iniciarse, el mapa de Uruguay. Permite agregar las ubicaciones de proveedores y demandantes (zonas afectadas que necesitan suministros), así como también especificar la cantidad de suministros disponibles y necesarios para las zonas proveedoras y demandantes respectivamente. Permite también especificar la cantidad de vehículos que se encuentran en los nodos proveedores al comienzo de la operación.

Se implementa un algoritmo (heurística) específicamente diseñado para resolver la distribución de suministros desde proveedores hacia demandantes ubicados geográficamente, en un tiempo de ejecución acotado, obteniendo resultados satisfactorios.

Finalmente, el resultado a nivel de software es un prototipo funcional, implementado utilizando mayoritariamente herramientas de software libre, facilitando la comprensión y modificación de código.

Los plazos del proyecto se fueron extendiendo a medida del desarrollo del mismo. La falta de experiencia de los integrantes del mismo en materia de investigación, llevó a que las estimaciones iniciales de los plazos no hayan sido del todo correctas.

A su vez, se invirtió mucho tiempo en la etapa de análisis e investigación para el proyecto, se relevaron y analizaron gran cantidad de artículos de alta complejidad matemática y algoritmos de difícil comprensión, realizando implementaciones y pruebas de los mismos en algunos casos.

En el proceso de desarrollo del prototipo de software, tampoco se pudieron cumplir con los plazos establecidos de antemano. Principalmente, las etapas de verificación y corrección de errores se extendieron más de lo previsto, debiendo realizar varias iteraciones entre correcciones y verificaciones (por “nuevos bugs” encontrados).

La investigación presentada en este proyecto es una invitación al desarrollo y promoción de la logística humanitaria en Uruguay dado que, hasta donde se ha podido indagar, no se aplica en el país.

7 TRABAJO FUTURO Y POSIBLES MEJORAS

Habiendo cumplido con el alcance propuesto para el proyecto, y obteniendo un software de calidad y funcional, dada la amplia variedad de aspectos que abarca la logística humanitaria, existen muchos puntos que podrían ser de interés analizar, estudiar y desarrollar en trabajos futuros, ya sea agregando nuevas funcionalidades al sistema así como también mejorando las ya existentes.

Un aspecto importante a mejorar del software es la fuente de datos que se utilizó. Para obtener el tiempo requerido en transportar suministros desde dos nodos (valores que se mantienen en el tiempo), se depende de fuentes externas cuando lo ideal sería que los datos estuvieran alojados localmente. Para lograr esto, se podrían cargar todas las localidades de Uruguay con los valores correspondientes, permitiendo que los tiempos se encuentren disponibles en la base de datos antes de que sea necesario calcular la distribución. Esto impactaría directamente en la performance del sistema en general mejorando los tiempos de ejecución y evitando el tener que lidiar con las desventajas del uso de herramientas hechas por terceros (actualmente Google Maps restringe la cantidad de nodos por consulta, no asegura devolver una respuesta para cualquier par de nodos, tiempos de respuesta dependientes de características de conexión a internet, etc). Previendo que una futura iteración de desarrollo del sistema utilice otra fuente de datos, se diseñó una arquitectura tal que, ante un cambio de esta magnitud, tenga el menor impacto sobre los distintos componentes del sistema.

Dentro de los puntos que no fueron manejados en el proyecto debido al alcance del mismo se encuentran algunas funcionalidades que podrían resultar de interés estudiar e implementar; por ejemplo podría implementarse un módulo que permita exportar la distribución solución obtenida a algún formato analítico (por ejemplo planillas de Excel), de manera de poder imprimirlo y distribuirlo a quien se considere necesario (por ejemplo choferes).

A su vez, también podría ser de interés el desarrollo de la funcionalidad que permita especificar calles en un mapa como “no transitables” (por ejemplo ArcGIS [62] lo permite), de manera que cuando se calcule la distribución solución estas calles no se tomen en cuenta (de gran utilidad considerando que se trabaja en contextos donde generalmente se puede ver afectada la infraestructura vial).

Dentro de las mejoras a realizarse en el sistema, podría considerarse el desarrollo de una heurística que resuelva el modelo planteado en el anexo [punto 8.1]; de esta manera, realizando pequeñas modificaciones dentro de la lógica e interfaz del sistema, se podrían manejar los suministros como elementos individuales, dotándolos de

atributos de interés como ser el nombre (arroz, agua, colchón, etc.), categoría (alimento, abrigo, herramienta, etc.), volumen, utilidad, etc.

A su vez, los vehículos asociados a los proveedores podrían contar con determinados atributos como capacidad (en volumen, por ejemplo metros cúbicos), consumo de combustible, costo fijo (por ejemplo mantenimiento y sueldo de choferes), etc.

Finalmente, también se sumaría la funcionalidad de poder mostrar el costo asociado a la solución obtenida. Este se obtiene principalmente de los costos asociados a los camiones utilizados (consumo y costo fijo). Además, también se permitiría calcular distribuciones con distintos niveles de prioridad (no atendiendo primero la demanda de los nodos más cercanos), priorizar cumplir la demanda de determinado/s nodo/s, o priorizar la entrega de determinado suministro dentro del sistema (por ejemplo insumos básicos).

Todas estas modificaciones, repercutirían en un sistema que representaría de manera más fiel la realidad, aunque podría verse perjudicado con tiempos de computo elevados para hallar las distribuciones solución, por eso resulta de gran importancia y se debe poner especial hincapié en el desarrollo de la heurística, buscando una solución de calidad pero a la vez que no requiera de demasiado tiempo para obtenerla.

Otra funcionalidad de interés podría ser la de implementar un sistema de avisos de envío y entrega de suministros, de esta manera cuando un camión con suministros sale de un nodo proveedor se avisa al nodo demandante (por ejemplo vía e-mail) que el mismo ha salido y la hora en la que aproximadamente arribe, a su vez el proveedor también es avisado cuando los suministros fueron entregados.

En relación a esto último, podría integrarse el sistema desarrollado a algún software de gestión y control de suministros web (por ejemplo que cuente con la funcionalidad mencionada anteriormente además de otras propiamente de la gestión de suministros). Uno de los candidatos podría ser el software Sahana Eden [punto 2.4.2], el cual cuenta con muchas funcionalidades (no solo de gestión de inventarios) y es de código abierto; claro que el modulo a integrar debe respetar los estándares provistos por Sahana, involucrando aspectos como el lenguaje a utilizarse, estilo de codificación, estilos de diseño, arquitectura, etc., por lo que podría ser una tarea no menor.

Si bien el sistema permite visualizar determinadas capas de datos (a través de servicios WMS), también podría ser de interés que los usuarios sean capaces de cargar diferentes capas de datos especificando la url con el servicio WMS que tenga dichos datos.

A su vez, también podría desarrollarse una funcionalidad que permita definir, por parte de expertos, las zonas afectadas y zonas que posiblemente sean afectadas con el pasar de los días (especificándose el impacto en orden cronológico si se desea). Estos datos pueden ser visibles a la población en general, ayudando a generar planes de evacuación preventivos a la vez que permiten tener una visión clara y definir la ubicación de los posibles refugios; vale aclarar que la API de Google Maps permite realizar polígonos

(definir zonas) sobre los mapas, por lo que su desarrollo no debería ser demasiado complejo. Estos datos de zonas afectadas también podrían obtenerse a partir de servicios WMS externos.

Finalmente, para un mejor y más ordenado manejo del sistema, podría resultar de interés definir distintos roles de usuarios, cada uno con sus respectivos permisos (realizando el control de acceso pertinente).

Por ejemplo, pueden haber roles asociados a cada nodo del mapa, de manera que cada uno de esos usuarios se encargará de mantener al día el inventario del depósito (en caso de que el usuario esté asociado a un nodo proveedor), o actualizar los suministros que hagan falta (en caso de que el usuario esté asociado a un nodo demandante); cada uno de estos usuarios solo podría modificar los atributos asociados a su nodo.

Por otra parte, si el sistema es accesible a la población en general, es lógico que se le restrinjan las funcionalidades asociadas a la gestión de suministros o hasta visualizar la distribución; sin embargo, podría ser de interés que se pudieran visualizar los refugios o centros médicos existentes (si existe la funcionalidad de ingresarlos). Además podrían ver los planes de evacuación existentes (de manera que los usuarios puedan ingresar al sistema y ver a dónde deben dirigirse), para esto es necesario definir zonas en el mapa que se encuentren asociadas a refugios, y a su vez se deben definir “camino seguros” (por ejemplo dos) desde la zona afectada (centro de la misma) hacia el refugio; desarrollar esta funcionalidad debería ser relativamente accesible utilizando las herramientas provistas por la API de Google Maps.

En conclusión, la logística humanitaria abarca una gran variedad de tareas asociadas a la fase de respuesta, y la utilización de un sistema de información geográfica nos abre un abanico de posibilidades muy grande en cuanto a la posibilidad de trabajar y representar datos geo-referenciados sobre un mapa, todo esto nos lleva a que existan muchísimas nuevas y variadas funcionalidades a agregarle al prototipo desarrollado que pudieran servir de interés a la logística humanitaria; lo que se intentó con el prototipo simplemente fue atacar uno de los puntos que hoy en día no se encuentra demasiado desarrollado (principalmente en lo referente a software libre) que es la distribución de suministros en un contexto de logística humanitaria.

8 ANEXO

8.1 MODELO DE DISTRIBUCIÓN (SUMINISTROS INDIVIDUALES)

El modelo que se desarrolla a continuación, parte del modelo explicado en el documento Estado del Arte (con dos niveles de prioridad); la gran diferencia con el mismo radica en que en el modelo original, los suministros se manejan como un conjunto o paquete, mientras que en el siguiente los mismo se toman como elementos individuales, pudiéndose trabajar con suministros particulares (por ejemplo arroz, agua, colchones, etc.) o gestionándolos en categorías (por ejemplo alimentos no perecederos, ropa, etc.). De esta manera, se logra una mayor aproximación a la realidad por parte del modelo, brindando también una mayor funcionalidad a la hora de la organización y gestión de suministros.

Por lo tanto, en cada nodo en el mapa se pueden especificar el tipo y la cantidad de suministros que tiene (si es proveedor) o necesita (si es demandante). Adicionalmente, también se puede especificar (opcionalmente) la cantidad de cada suministro que se desea distribuir en la operación (pueden existir escenarios donde se cuente con la cantidad de suministros necesarios como para satisfacer la demanda existente, y sin embargo decidir no distribuir todos los suministros en una sola vez, sino que distribuirlos en etapas sucesivas por ejemplo).

Finalmente, considerando que es posible especificar la cantidad de cada suministro que se desea distribuir en la operación, también es posible ponderar los suministros entre sí, logrando que la distribución de determinados suministros sea “más importante” que la de otros suministros (por ejemplo productos básicos).

Aclaremos que no se profundizará demasiado en la explicación de los puntos que se tienen en común con el modelo original [Estado del Arte – punto 4.2]. A su vez, las restricciones asociadas a la probabilidad de saqueo y confiabilidad de las rutas fueron eliminadas por considerarse no muy relevantes ni aplicables en la región.

Datos que forman parte de la estructura del modelo:

- $N = \{1, \dots, n\}$: conjunto de nodos (demandantes, proveedores e intermedios)
- A : conjunto de aristas entre los nodos i y j , siendo $i, j \in N$.
- l_{ij} : distancia del camino entre el nodo i y j siendo $(i, j) \in A$.
- vel_{ij} : velocidad promedio que se puede desarrollar en la arista (i, j) .
- W : cantidad de suministros diferentes existentes.
- vol_w : volumen de una unidad del suministro w . Pueden ser definido en

Kilogramos, metros cúbicos (viendo el volumen que ocupa) o la unidad que se quiera (pero todos los suministros deben utilizar la misma unidad).

- s_{iw} : cantidad de suministros del tipo “w” en el nodo $i \in N$ (medido en la unidad que se utilice, por ejemplo Kg).
- d_{iw} : cantidad de suministros del tipo “w” demanda en el nodo $i \in N$ (medido en la unidad que se utilice, por ejemplo Kg).
- q_w : cantidad de suministros del tipo “w” a ser distribuida en la operación (medido en la unidad que se utilice, por ejemplo Kg).
- $pond1_w$: penalización por apartarse de los objetivos q_w . Se logra una ponderación para los distintos tipos de suministros (sirve para indicar qué suministros son más importantes distribuir [suministros básicos]).
- $pond2_{obj}$: penalización por apartarse de los objetivos pertenecientes al segundo nivel.
- K : cantidad de tipos de vehículos diferentes existentes.
- e_k : capacidad del vehículo de tipo k (medido en la unidad que se utilice, por ejemplo Kg).
- cv_{ij} : costo de transportar mercadería desde el nodo i al j, siendo $(i,j) \in A$ (costo por unidad (ejemplo Kg) y distancia).
- cf_{kij} : costo de reparación de un vehículo del tipo k utilizado en el camino (i,j) (costo por unidad de distancia).
- a_{ki} : disponibilidad de vehículos del tipo k en el nodo $i \in N$.
- b : presupuesto de la operación.

Variables de decisión:

- $LORRY_{ijk}$: número de vehículos del tipo k usados en el link $(i,j) \in A$
- $LOADTR_{ijw}$: cantidad de suministros del tipo w llevados desde i a j
- $LOAD_{iw}$: cantidad de suministros del tipo w que se encuentran en el nodo i al final de la operación.
- USE_{ij} : variable binaria que tiene el valor 1 si algún vehículo está viajando del nodo i al j, sino vale 0
- Z_i : variable continua para la eliminación de sub-ciclos
- Yq_w : variable de desviación de los objetivos q_w . Indica las diferencias existentes entre los niveles de aspiración impuestos por el decisor y el resultado que se alcanzan en el objetivo.

Restricciones duras (de obligatorio cumplimiento):

- Correspondencia de carga en cada nodo:

$$\sum_{i/(j,i) \in A} LOADTR_{jiw} - \sum_{i/(i,j) \in A} LOADTR_{ijw} = s_{jw} - LOAD_{jw}, \forall j \in N, \forall w \in W$$

(1)
- Límite de transporte de suministros a un nodo:

$$LOAD_{iw} \leq d_{jw}, \forall j \in N, \forall w \in W, d_{jw} > 0$$

(2)

- Correspondencia de vehículos en cada nodo:

$$\sum_{i/(j,i) \in A} LORRY_{jik} - \sum_{i/(i,j) \in A} LORRY_{ijk} = a_{kj}, \forall j \in N, \forall k \quad (3)$$

- Relación vehículos-carga (limita la carga transportada a la capacidad del vehículo):

$$LOADTR_{ijw} \cdot vol_w \leq \sum_k e_k LORRY_{ijk}, \forall (i,j) \in A, \forall w \in W \quad (4)$$

- Relación vehículos-carga-uso (permite utilizar vehículos vacíos entre dos nodos solamente si luego serán utilizados para levantar suministros de un nodo y llevarlos a otro).

$$\sum_k e_k LORRY_{ijk} - \max_k(e_k) \leq LOADTR_{ijw}, \forall (i,j) \in A, \forall w \in W \quad (5)$$

- Eliminación de sub-ciclos (formulación de Miller-Trucker-Zemlin [14]):

$$Z_i - Z_j + n \cdot USE_{ij} \leq n - 1, \forall i, j \in N, i \neq j, Z_i \in R \quad (6)$$

- Relación vehículos-uso:

$$\sum_k LORRY_{ijk} \leq USE_{ij} \sum_k (\sum_{l \in N} a_{kl}), \forall (i,j) \in A \quad (7)$$

$$USE_{ij} \leq \sum_k LORRY_{ijk}, \forall (i,j) \in A \quad (8)$$

- Presupuesto (el modelo asume que los vehículos retornan vacíos por el mismo camino que por el que fueron):

$$\sum_{(i,j) \in A} l_{ij} \cdot (cv_{ij} \cdot \sum_w (LOADTR_{ijw} \cdot vol_w) + \sum_k 2 \cdot cf_{kij} \cdot LORRY_{ijk}) \leq b \quad (9)$$

- Cumplimiento de la demanda (primeramente se busca satisfacer la demanda de los suministros con mayor ponderación)

$$\sum_{i \in N/d_{iw} > 0} LOAD_{iw} + Yq_w = q_w, \forall w \in W \quad (10)$$

Como en el modelo original, en este también existen dos niveles de prioridad, el primero es el que busca cumplir (en la mayor medida posible) con los objetivos q_w (cantidad de suministros a distribuir). Luego, en el segundo nivel se definen objetivos opcionales referentes a factores de ruteo, como ser ajustar del costo al presupuesto, tiempo en recorrer el camino; y referentes a factores de demanda, como ser equitatividad en la distribución de suministros y prioridad de determinados nodos para recibir suministros, pudiendo definir niveles de aspiración para cada uno de dichos objetivos y prioridades entre ellos.

Finalmente, planteamos el modelo a resolver en el primer nivel.

$$\begin{aligned}
& \min \sum_w (Yq_w \cdot pondSum_w) \\
& s. t. \\
& restricciones (1) - (10) \\
& LORRY_{ijk}, LOADTR_{ijw}, LOAD_{iw} \geq 0 \text{ y enteros} \\
& USE_{ij} \in \{0,1\}, \quad Yq_w \geq 0
\end{aligned}$$

Para el segundo nivel, se define la variable t_{obj} (una variable para cada objetivo del segundo nivel), la cual representa los niveles de aspiración de los objetivos opcionales mencionados anteriormente (ajuste del presupuesto, priorizar suministros a determinados nodos, etc.).

De esta manera, se definen las siguientes variables asociadas a los objetivos:

- Y_{obj} : variable de desviación para el objetivo t_{obj} , estas variables son positivas por definición.
- $DMAX$: máxima desviación de la demanda de los nodos

Restricciones blandas (de cumplimiento optativo):

Estas restricciones serán consideradas teniendo en cuenta la ponderación que el decisor resuelva sobre cada una de ellas.

- Se mantiene la solución óptima obtenida en el primer nivel (Yq_w^*), y en base a esta se calculan las metas del segundo nivel. (Esta es una restricción que debe cumplirse).

$$\sum_{i \in N/d_{iw} > 0} LOAD_{iw} + Yq_w^* = q_w, \quad \forall w \in W \quad (11)$$

- Restricción asociada al costo de la operación:

$$\sum_{(i,j) \in A} l_{ij} (cv_{ij} \sum_{w \in W} (LOADTR_{ijw} \cdot vol_w) + \sum_k 2 \cdot cf_{kij} \cdot LORRY_{ijk}) - Y_1 \leq t_1 \quad (12)$$

- Restricción para ajustar el tiempo en el que se deben distribuir los suministros:

$$\sum_{(i,j) \in A} ((l_{ij}/vel_{ij}) \cdot \sum_{w \in W} LOADTR_{ijw}) - Y_2 \leq t_2 \quad (13)$$

- Restricciones para lograr una distribución equitativa de los suministros entre los demandantes (no necesariamente abastecer primero a los más cercanos):

$$1 - \sum_{w \in W} \frac{LOAD_{iw}}{d_{iw}} \leq DMAX, \quad \forall i \in N/d_{iw} > 0 \quad (14)$$

$$DMAX - Y_3 \leq t_3 \quad (15)$$

- Restricción para priorizar la entrega de suministros a determinado/s nodo/s:

$$\sum_{w \in W} \frac{LOAD_{Lw}}{d_{Lw}} + Y_i \geq t_i \quad (16)$$

Siendo L el nodo priorizado. La cantidad de restricciones de este tipo son variables, debiéndose definir una de estas restricciones por cada nodo a priorizar.

Considerando los objetivos y variables planteadas anteriormente, lo que se busca en el segundo nivel es determinar un nivel de aspiración para cada objetivo y minimizar las desviaciones de dichos niveles; de esta manera, si consideramos todos los objetivos como un conjunto, lo que buscaremos será minimizar la suma ponderada de las desviaciones respecto a los niveles de aspiración.

Finalmente, planteamos el modelo perteneciente al segundo nivel.

$$\begin{aligned}
 & \min \sum_{obj} pond2_{obj} \cdot Y_{obj} \quad (obj \text{ es el número de objetivos del segundo nivel}) \\
 & s. t. \\
 & (1) - (9), (11) - (16) \\
 & LORRY_{ijk}, LOADTR_{ijw}, LOAD_{iw} \geq 0 \text{ y enteros} \\
 & USE_{ij} \in \{0,1\}, Y_{obj} \geq 0
 \end{aligned}$$

9 REFERENCIAS

- [1] Procedimientos Metaheurísticos en Optimización Combinatoria; R. Martí; Departamento de Estadística e Investigación Operativa – Facultad de Matemáticas, Universidad de Valencia; Matemàtiques, vol. 1; 2003; pp. 23-32; <http://www.uv.es/~rmarti/paper/docs/heur1.pdf>; último acceso 8/9/2013
- [2] Java Features; D. Kumar; 2008; <http://www.roseindia.net/java/java-introduction/java-features.shtml>; último acceso 8/9/2013
- [3] Features of Java; <http://ecomputernotes.com/java/what-is-java-evolution/explain-features-of-java>; último acceso 8/9/2013
- [4] Arquitectura y Diseño de Aplicaciones Java EE; C. Gavidia; <http://www.slideshare.net/cptanalatriste/arquitectura-y-diseo-de-aplicaciones-java-ee>; último acceso 8/9/2013
- [5] Java EE; https://es.wikipedia.org/wiki/Java_EE; último acceso 8/9/2013
- [6] Java SE Technologies – Database; <http://www.oracle.com/technetwork/java/javase/jdbc/index.html>; último acceso 8/9/2013
- [7] Enterprise JavaBeans Technology; <http://www.oracle.com/technetwork/java/javaee/ejb/index.html>; último acceso 8/9/2013
- [8] Java Persistence API; <http://www.oracle.com/technetwork/java/javaee/tech/persistence-jsp-140049.html>; último acceso 8/9/2013
- [9] JavaServer Pages Overview; <http://www.oracle.com/technetwork/java/overview-138580.html>; último acceso 8/9/2013
- [10] MySQL Workbench 5.2 - Visual Database Design; <http://www.mysql.com/products/workbench/design/>; último acceso 8/9/2013
- [11] What is jQuery and How to Start using jQuery; S. Narayan; 2011; <http://www.codeproject.com/Articles/157446/What-is-jQuery-and-How-to-Start-using-jQuery>; último acceso 8/9/2013
- [12] Introducing JSON; <http://www.json.org/>; último acceso 8/9/2013

- [13] Arquitectura J2EE;
<http://users.dcc.uchile.cl/~jbarrios/J2EE/node14.html>; último acceso 8/9/2013
- [14] A note on the lifted Miller–Tucker–Zemlin subtour elimination constraints for the capacitated vehicle routing problem; I. Kara, G. Laporte, T. Bektas; 2004; European Journal of Operational Research, Vol. 158, no. 3; DOI 10.1016/S0377-2217(03)00377-1; pp. 793-795
- [15] Understanding Java Platform, Enterprise Edition;
<http://docs.oracle.com/javaee/6/firstcup/doc/gkhoy.html>; último acceso 8/9/2013
- [16] Assembla Workspace;
<https://www.assembla.com/workspaces?>; último acceso 8/9/2013
- [17] LSS - Sistema de Apoyo Logístico - Manual de Usuario y Prácticas Módulo Windows; Organización Panamericana de la Salud - Área de Preparativos para Situaciones de Emergencia y Socorro en Caso de Desastre; 2006;
<http://preparativosyrespuesta.cridlac.org/XML/txt/doc90793/doc90793-contenido.pdf>; último acceso 8/9/2013
- [18] GLPK (GNU Linear Programming Kit);
<http://www.gnu.org/software/glpk/>; último acceso 8/9/2013
- [19] LSS home page; <http://www.lssweb.net/>; último acceso 8/9/2013
- [20] LSS - Logistics Support System; <http://www.itu.int/ITU-D/emergencytelecoms/events/Alexandriaconference/presentations/Doc41-Fundesuma.pdf>; último acceso 8/9/2013
- [21] Sahana Eden; <http://sahanafoundation.org/products/eden/>; último acceso 8/9/2013
- [22] HELIOS Software; <http://www.humanitarianlogistics.org/helios-software>; último acceso 8/9/2013
- [23] Collabit; <https://code.google.com/p/collabbit/>; último acceso 8/9/2013
- [24] Posit; <http://posit.hfoss.org/about>; último acceso 8/9/2013
- [25] Disaster Management Information System (DMIS); Le Minh Ba; Disaster Management Working Group meeting; 2009;
<http://www.slideshare.net/ict4devwg/presentation-on-disaster-management-information-system-by-le-minh-ba-ndmp>; último acceso 8/9/2013
- [26] Disaster Management Information System – Project Document; AIMS Project

- Team; 2004;
http://www.aims.org.af/about/proj_reports/sub_project_docs/dmis_proj_scope.pdf; último acceso 8/9/2013
- [27] MySQL Server
<http://dev.mysql.com/downloads/mysql/5.5.html>; último acceso 8/9/2013
- [28] A lexicographical goal programming based decision support system for logistics of Humanitarian Aid; J. Montero, T. Ortuño, J. Rodríguez, G. Tirado, B. Vitoriano; 2010; TOP, 2011, Vol. 19, no. 2; ISSN 1863-8279; DOI 0.1007/s11750-010-0138-8; Ed. Springer US; pp. 464-479
- [29] Heurística para la Generación de un Conjunto de Referencia de Soluciones que Resuelvan el Problema de Ruteo de Vehículos con Múltiples Depósitos MDVRP; J. López, S Nieto; 2012; Tenth LACCEI Latin American and Caribbean Conference (LACCEI 2012) - Building Infrastructure by fostering engineering collaboration, efficient and effective integration and innovative planning (2012 - ciudad de Panama); <http://www.laccei.org/LACCEI2012-Panama/RefereedPapers/RP029.pdf>; último acceso 8/9/2013
- [30] Sistema de Manejo Integral de Desastres; pp. 1-11, 19-32, 50-51, 64-92; <http://www.medynet.com/usuarios/jraguilar/manejo.pdf>; último acceso 8/9/2013
- [31] SINAIE - Ley N°18621;
<http://www.parlamento.gub.uy/leyes/AccesoTextoLey.asp?Ley=18621&Anchor=>
; último acceso 8/9/2013
- [32] Guía Municipal para la Gestión del Riesgo; R. Vargas, J. Bárcenas, A. García, D. Peña, G. Coral, H. Gómez, I. Caicedo; 2010; Proyecto de asistencia técnica en gestión local del riesgo a nivel municipal y departamental en Colombia - Programa de Reducción de la Vulnerabilidad Fiscal del Estado frente a Desastres Naturales; pp. 5-44;
<http://www.boyaca.gov.co/index.php?idcategoria=18875&download=Y>; último acceso 8/9/2013
- [33] Capacity Building in Asia using Information Technology Applications; Disaster Managment, module 4; <http://www.adpc.net/casita/course-materials/Mod-4-Disaster-Mgmt.pdf>; último acceso 8/9/2013
- [34] Etapas del ciclo de los Desastres;
<http://desastres.usac.edu.gt/documentos/pdf/spa/doc12256/doc12256-contenido.pdf>; último acceso 8/9/2013
- [35] Ciclo de los Desastres; Seminario de Capacitación a Comités Distritales – Managua;
<http://desastres.usac.edu.gt/documentos/pdf/spa/doc16428/doc16428-1.pdf>;
último acceso 8/9/2013

- [36] La Gestión del Riesgo de Desastres: un Enfoque Basado en Procesos; L. Narváez, A. Lavell, G. Pérez; 2009; Proyecto “Apoyo a la Prevención de Desastres en la Comunidad Andina (PREDECAN)”; ISBN 978-9972-787-88-1; Ed. Secretaría General de la Comunidad Andina (Lima - Perú); pp. 9-83; http://www.comunidadandina.org/predecan/doc/libros/PROCESOS_ok.pdf; último acceso 8/9/2013
- [37] Logística y tecnología en la Acción Humanitaria; R. Carrasco, J. Bará, P. Brunet; capítulo 10 del libro “Tecnologías para el Desarrollo de las Comunidades Aisladas”; Ed. Real Academia de Ingeniería; ISBN 978-84-95662-44-6; pp. 427-447; 2011; <http://www.raing.es/sites/default/files/CAP%C3%8DTULO%2010%20-%20CRA.pdf>; último acceso 8/9/2013
- [38] Control de Inventarios Investigación de Operaciones 4; M.J. Moya Navarro pp. 19-25; http://books.google.es/books?id=uG8_nuimuhAC&printsec=frontcover&hl=e#v=onepage&q&f=false; último acceso 8/9/2013
- [39] Introduction to Logistics Systems Planning and Control; G. Ghiani, G. Laporte and R. Musmanno; 2004; paginas 14-18, 103-137; ISBN 9780470014042; DOI 10.1002/0470014040; Eds. S. Ross, R. Weber; http://www.pc-freak.net/international_university_college_files/Introduction%20to%20Logistic%20Systems%20Planning%20&%20control.pdf; último acceso 8/9/2013
- [40] Logistics, Disaster Management and Training Program; R. Stephenson; 1993
- [41] Gestión del Riesgo de Desastres – Para la Planificación del Desarrollo Local; L. Cosamalón, V. Lovón, M. Sicre, A. Vilela, L. Málaga, G. Masana, A. Montesinos, J. Ruiz; 2009; Proyecto “Fortaleciendo la participación social y las capacidades de gestión de los gobiernos locales en el proceso de reconstrucción”; Ed. Mesa de Concertación para la Lucha contra la Pobreza y Cáritas del Perú; pp. 13-48; <http://bvpad.indeci.gob.pe/doc/pdf/esp/doc1369/doc1369.pdf>; último acceso 8/9/2013
- [42] Página oficial de SourceForge; <http://sourceforge.net/>; último acceso 01/8/2013
- [43] Guía para el Establecimiento de Refugios Temporales; Dirección Nacional de Protección Civil del Estado de San Luis de Potosí; pp. 4-19; [http://201.144.123.34/webproteccion.nsf/001162fdcf442a3a0625770500632c41/\\$FILE/ATTV90UY/Manual%20de%20Refugios%20Temporales.pdf](http://201.144.123.34/webproteccion.nsf/001162fdcf442a3a0625770500632c41/$FILE/ATTV90UY/Manual%20de%20Refugios%20Temporales.pdf); último acceso 8/9/2013
- [44] Algoritmos Genéticos Aplicados al Diseño de una Red de Comunicaciones Confiable; D. Calegari; 2002; Proyecto de Grado (Facultad de Ingeniería – Universidad de la Republica – Uruguay); pp. 13-19;

<http://www.fing.edu.uy/inco/pedeciba/bibliote/tgradoinf/tg-calegari.pdf>; último acceso 8/9/2013

- [45] Guía para la elaboración - Plan de Emergencias; pp. 8-10; http://www.tramites.go.cr/manual/espanol/10/plan_emergencia.pdf; último acceso 8/9/2013
- [46] Programación por Metas (Goal Programming): pasado, presente y futuro; C. Romero; 2002; Revista Electrónica de Comunicaciones y Trabajos de ASEPUMA; ISBN 9788484426301; Ed. Tirant Lo Blanch; pp. 75-89; http://www.uv.es/sala/malaga/recta/extraordinarios/Vol_01/04t.pdf; último acceso 8/9/2013
- [47] Multiperiod integrated routing and scheduling of World Food Programme cargo planes in Angola; V. De Angelisa, M. Melcolia, C. Nikoib, G. Storchia; 2007; Computers & Operations Research, Vol. 34, no. 6; Ed. Elsevier; DOI 10.1016/j.cor.2005.07.012; pp. 1601-1615; <http://www.sciencedirect.com/science/article/pii/S030505480500239X>; último acceso 8/9/2013
- [48] A multi-objective robust stochastic programming model for disaster relief logistics under uncertainty; A. Bozorgi-Amiri, M. Jabalameli, S. Mirzapour Al-e-Hashem; 2011; OR Spectrum; ISSN 1436-6304; DOI 10.1007/s00291-011-0268-x; Ed. Springer-Verlag; pp. 640-644
- [49] Humanitarian Relief Chains: Issues and Challenges. BEAMON, B.M. En Proceedings of the 34th International Conference on Computers and Industrial Engineering, Vol. 34, pp. 77-82, 2004; <http://www.docstoc.com/docs/31881727/HUMANITARIAN-RELIEF-CHAINS>, último acceso 8/9/2013
- [50] Sistema de Manejo de Suministros Humanitarios (SUMA); <http://www.disaster-info.net/SUMA/spanish/index.htm>; último acceso 8/9/2013
- [51] Cómo elaborar un Plan de Evacuación; Dpto. Instrucción Técnica de Prevención – Dirección de Educación Civil - Ministerio de Seguridad Dirección General de Defensa Civil de la Provincia de Buenos Aires; <http://www.mseg.gba.gov.ar/defensacivil/educacivil/plan%20de%20evacuacion.pdf>; último acceso 8/9/2013
- [52] API de Google Maps; <https://developers.google.com/maps/?hl=es>; último acceso 8/9/2013
- [53] Teórico de Arquitecturas Empresariales, curso de Taller de Sistemas de Información Geográficos Empresariales; InCo Facultad de Ingeniería UDELAR; Diapositivas: 72-79; Web <https://eva.fing.edu.uy/>; Último acceso: 8/9/2013

- [54] A Stochastic Humanitarian Inventory Control Model for Disaster Planning
Autores: K. Ozbay, E. Ozguven; 2006; Transportation Research Record: Journal of
the Transportation Research Board, No. 2022; ISSN: 0361-1981; pp. 63-75;
<http://www.rits.rutgers.edu/files/disasterplanning.pdf>; último acceso 8/9/2013

- [55] Scheduling of vehicles from a central depot to a number of delivery points; G.
Clarke, W. Wright; 1962; Operations Research, Vol. 12, no. 4; DOI
10.1287/opre.12.4.568; pp. 568-581;
<http://read.pudn.com/downloads160/doc/fileformat/721736/Scheduling%20of%20vehicles%20from%20a%20central%20depot%20to%20a%20number%20of%20delivery%20points.pdf>; último acceso 8/9/2013

- [56] Métodos Exactos y Heurísticos para resolver el Problema del Agente Viajero y el
Problema de Ruteo de Vehículos; F. Sandoya; 2007; Decimocuarta Jornada en
Estadística e Informática – Ecuador;
http://www.icm.espol.edu.ec/jornadas/14/archivos/Diapositivas/SandoyaFernando/conferencia/SandoyaFernando_M%C3%A9todos_exactos_y_heur%C3%ADsticos_para_el%20VRP_jornadas.pdf; último acceso 8/9/2013

- [57] Algorithms for the vehicle routing and scheduling problem with time window
constraints; M. Solomon; 1987; Operations Research, Vol. 35, no. 2; DOI
10.1287/opre.35.2.254; pp. 254-265;
http://www.iro.umontreal.ca/~dift6751/paper_solomon.pdf; último acceso 8/9/2013

- [58] The Google Distance Matrix API;
<https://developers.google.com/maps/documentation/distancematrix/>; último acceso 8/9/2013

- [59] Google Code;
<https://code.google.com/intl/es-419/>; último acceso 01/8/2013

- [60] Página de Glasfish;
<https://glassfish.java.net/es/>; último acceso 01/8/2013

- [61] Página de Oracle para descargar componentes de Java y Netbeans;
<http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>;
último acceso 01/8/2013

- [62] ArcGIS; <http://www.arcgis.com/about/features.html>; último acceso 8/9/2013

- [63] SINAE - Catálogo Histórico de Desastres del Uruguay;
http://www.sne.gub.uy/index.php?option=com_content&view=article&id=230&Itemid=87; último acceso 30/5/2013

Estado del arte

Proyecto de grado Logística Humanitaria

Integrantes

Martín Fernández Staehle
Laura Aizpún Fernández

Tutores

Omar Viera
Sandro Moscatelli

Departamento de Investigación Operativa

Instituto de Computación

Facultad de Ingeniería

Universidad de la República Oriental del Uruguay



CONTENIDO

1	Gestión de Riesgos.....	4
1.1	Introducción	4
1.2	Definiciones	5
1.2.1	Desastres y tipos de desastres	5
1.2.2	Vulnerabilidad, amenaza y riesgo	5
1.3	Proceso	6
1.3.1	Fase anterior a la ocurrencia de un desastre	7
1.3.1.1	Prevención.....	7
1.3.1.2	Mitigación	8
1.3.1.3	Preparación.....	8
1.3.1.4	Alerta.....	9
1.3.2	Fase posterior a la ocurrencia de un desastre	9
1.3.2.1	Respuesta.....	9
1.3.2.2	Manejo de Suministros	10
1.3.2.3	Rehabilitación	10
1.3.2.4	Reconstrucción.....	11
1.4	SINAE	11
2	Logística	13
2.1	Que es logística.....	13
2.2	Logística humanitaria	13
2.3	Diferencias entre logística humanitaria y logística comercial.....	16
3	Gestión de inventarios	18
3.1	Introducción	18
3.2	Aplicación al caso de estudio.....	18
3.3	Gestión de inventarios: Relevamiento de costos	19
3.4	Modelos de gestión de inventarios.....	19
3.4.1	Generalidades	19
3.4.2	Único punto de almacenamiento, un solo ítem y demanda constante	20
3.4.3	Modelo con múltiples ítems	22
4	Distribución de Suministros	24
4.1	Introducción	24
4.2	Descripción del Modelo.....	26

4.3	Implementación del Modelo	32
5	Evacuaciones.....	34
5.1	Introducción	34
5.2	Evacuaciones de estructuras	34
5.3	Evacuación del área urbana.....	35
5.3.1	Plan de Evacuación - zonas urbanas.....	36
6	Albergues Temporales.....	39
6.1	Introducción	39
6.2	Modelos matemáticos	42
7	Relevamiento de Heurísticas y Metaheurísticas para la Distribución de Suministros	47
7.1	TSP.....	47
7.2	VRP.....	48
7.2.1	Heurísticas.....	48
7.2.1.1	Algoritmo de Ahorros de Clarke y Wright.....	49
7.2.1.2	Heurísticas de Inserción	49
7.2.1.3	Cluster first - Route second.....	49
7.2.1.4	Route first - Cluster second.....	50
7.2.1.5	Búsqueda Local	51
7.2.2	Metaheurísticas.....	52
7.2.2.1	Optimización basada en Colonia de Hormigas.....	53
7.2.2.2	GRASP	54
7.2.2.3	Búsqueda Tabú.....	55
7.2.2.4	Algoritmos Genéticos	55
8	Relevamiento de software cartográfico	57
8.1	Sistemas de información geográfica	57
8.1.1	GvSIG dektop:.....	57
8.1.2	Google Maps.....	58
8.1.3	Bing Maps.....	59
8.1.4	ArcGis.....	59
9	Relevamiento de solvers para problemas de programación MIP.....	60
9.1	Problemas de programación MIP.....	60
9.2	Solvers.....	60
9.2.1	GAMS	60
9.2.2	SCIP	60
9.2.3	GLPK.....	61
10	Anexo.....	62

10.1	Anexo I: Software SUMA	62
10.2	Anexo II: Programación por Metas	65
11	Referencias	67

1 GESTIÓN DE RIESGOS

1.1 INTRODUCCIÓN

La Gestión de Riesgos es la disciplina que se encarga de la gestión de desastres, debiendo administrar los recursos existentes de modo de poder actuar ante la ocurrencia de un desastre teniendo como objetivo principal el aspecto humanitario. En consecuencia, sus funcionalidades comprenden todo lo referente a la preparación antes de que ocurra el desastre, actuar cuando el mismo suceda, y luego de que ocurre, suministrar ayuda a la zona afectada [8].

Complementando lo anterior, la gestión de riesgos consiste en un conjunto de procedimientos y actividades que son planificadas y llevadas a cabo con el fin principal de reducir el impacto del desastre en la zona afectada (principalmente en lo referente al factor humanitario aunque también considerando otros factores como infraestructura y servicios) y disminuir en la medida de lo posible la probabilidad de ocurrencia del mismo [5].

Este proceso puede dividirse fundamentalmente en dos fases [6]:

1. Antes que ocurra el desastre, donde se llevan a cabo tanto actividades relacionadas a disminuir (en la medida de lo posible) la probabilidad de ocurrencia del desastre, como actividades destinadas a preparar una adecuada respuesta en caso de que el desastre ocurra, asegurando una respuesta adecuada.
2. Después de ocurrido el desastre, donde se llevan a cabo actividades de respuesta, cuyo principal objetivo es llevar ayuda a las zonas afectadas, disminuyendo en lo posible las pérdidas de vidas humanas y logrando aliviar el sufrimiento de la población afectada. Además, una vez atendidas las necesidades más urgentes de la población, se deben llevar a cabo planes que permitan la reconstrucción y recuperación de la sociedad, realizando intervenciones con el fin de recuperar gradualmente el nivel de actividad social y económico perdido, reconstruyendo la infraestructura dañada, y restableciendo los servicios básicos que puedan haber quedado colapsados luego de ocurrido el desastre.

En definitiva, teniendo en cuenta lo mencionado anteriormente, la gestión de riesgos, es una disciplina encargada de la planificación, la organización, la dirección y el control de las actividades realizadas en cada una de las fases del proceso citado [5]. Más adelante se profundizará en el proceso mencionado previamente.

1.2 DEFINICIONES

1.2.1 DESASTRES Y TIPOS DE DESASTRES

Un desastre es definido en [5] como un suceso repentino que causa una interrupción grave en el funcionamiento de una comunidad o sociedad, pudiendo provocar pérdidas humanas, personas heridas, deterioro de infraestructura y servicios, en una medida tal, que la comunidad afectada no pueda afrontar dicha situación con los recursos locales, necesitando ayuda externa.

Durante mucho tiempo (aproximadamente hasta la época de los '70), los desastres se clasificaban principalmente en dos categorías: los naturales y los causados por el hombre; donde los desastres naturales los constituían por ejemplo las inundaciones, terremotos, y eventos similares, mientras que los desastres provocados por el hombre principalmente se identificaban como accidentes industriales y similares [5].

Esta percepción de la realidad, un tanto simplista por cierto, hoy en día ya no se adecuaba a la realidad que conocemos. La creciente ocurrencia de desastres naturales y el aumento de la ferocidad con la que ocurren, han llevado a realizar numerosos estudios respecto a qué es lo que está causando la ocurrencia de dichos eventos y por qué se presentan con tal magnitud. En consecuencia, hoy en día se sabe que gran cantidad de los desastres que antes se conocían como “desastres naturales” se encuentran fuertemente influenciados por la actividad humana [8]. Por ejemplo, a nivel mundial, científicos aseguran que las olas de fríos sufridas en Europa, junto a otras varias distorsiones climáticas en el resto del planeta, se deben en gran medida al calentamiento global; otros ejemplos de la intervención humana pueden ser la ocurrencia de inundaciones por la tala indiscriminada de árboles, eliminando una “barrera natural” contra inundaciones, o la aparición de la gripe aviar, producto de un conjunto de errores humanos. Por ende, queda claro que en los últimos años las actividades humanas en muchos casos han influido negativamente en los efectos de los fenómenos naturales, ampliando la vulnerabilidad para la población.

Por otra parte, si bien la magnitud del desastre puede determinar la severidad del impacto sobre la zona, cabe destacar que en muchas ocasiones son las condiciones de vulnerabilidad en las que se encuentra una comunidad (más que la amenaza en sí) las que determinan la magnitud del daño y por ende las responsables de las pérdidas humanas [8].

1.2.2 VULNERABILIDAD, AMENAZA Y RIESGO

Se define una amenaza o peligro a la probabilidad de ocurrencia de un evento externo a la comunidad, que en caso de manifestarse puede producir un gran desastre a dicha comunidad [8].

A partir de lo anterior, se establece la vulnerabilidad de una comunidad como un factor intrínseco de la misma, que indica su grado de exposición frente a la potencial ocurrencia de un determinado desastre. Se refiere a un conjunto de características propias de la comunidad, que predisponen a la misma a sufrir daños graves de concretarse una amenaza [8].

Finalmente, el riesgo de una comunidad a sufrir un desastre se define como la estimación de probables pérdidas a nivel humanitario, de infraestructuras, económico, etc., y se calcula en función de la relación existente entre la gravedad de la potencial amenaza y las condiciones de vulnerabilidad existentes [8].

1.3 PROCESO

Como se mencionó anteriormente, ésta sección del documento se centrará básicamente en un proceso para la gestión de riesgos que se basa en un proceso secuencial, basado en el trabajo antes e inmediatamente después de que ocurra el desastre (modelo tradicional [11]).



Figura 1 - Ciclo Gestión de Riesgos [92]

En la fase anterior a la ocurrencia del desastre se desarrollan las siguientes etapas [6]:

- Prevención
- Mitigación
- Preparación
- Alerta

Mientras que en la fase posterior al desastre se llevan a cabo las siguientes etapas [6]:

- Respuesta
- Rehabilitación
- Reconstrucción

Vale aclarar, que también existe otro modelo, llamado modelo de expansión-contracción, donde se consideran la gestión de riesgos como un proceso continuo, donde existe una serie de actividades que pueden llevarse en paralelo, aumentando o disminuyendo su acción en función del tiempo y dependiendo de la situación y el contexto en el que se encuentre la comunidad afectada. A modo de ejemplo, se pueden llevar en paralelo las etapas de prevención y mitigación con las etapas de reconstrucción y rehabilitación, aunque obviamente la intensidad de cada una varía con el correr del tiempo [4] [93].

Además, también existen modelos que se enfocan en las vulnerabilidades y el contexto en el que se encuentra inmersa la región, teniendo como objetivo la reducción de riesgos. Un ejemplo de estos son los modelos de Disaster Crunch y Release Models, los cuales buscan comprender y explicar las causas de los desastres, utilizando una perspectiva de causa-efecto [4].

A continuación se detallará en qué consiste cada una de las etapas mencionadas en el modelo tradicional planteado anteriormente junto con algunas de las actividades que son llevadas a cabo.

1.3.1 FASE ANTERIOR A LA OCURRENCIA DE UN DESASTRE

Desde hace muchos años, se vienen estudiando e intentando reducir los problemas relacionados a las vulnerabilidades de las comunidades frente a la ocurrencia de desastres, como ser construcciones inadecuadas en zonas de terremotos, asentamiento de comunidades en zonas inundables, etc. [9].

Lamentablemente, los costos relacionados a la disminución de las vulnerabilidades ya presentes son muy altos y no siempre se pueden afrontar. Sumado a esto, cada vez se registran más desastres a lo largo del mundo (principalmente en países con escasos recursos, donde las vulnerabilidades de las comunidades son extremadamente altas y no se las puede combatir debido a la escasez de dinero). Éste es un fenómeno que tiende a acentuarse consecuencia del cambio climático (entre otros factores). Todas estas condiciones, entre otras, llevan a que hoy en día, se haga gran hincapié en los procedimientos y metodologías aplicadas previas a la ocurrencia de un desastre con el fin de reducir sus efectos [9].

1.3.1.1 PREVENCIÓN

La prevención tiene por objetivo determinar posibles amenazas referentes a una determinada zona, y detectar las vulnerabilidades presentes en la comunidad en caso de concretarse dichas amenazas, con el fin de calcular el riesgo esperado [11] [6]. De esta manera, se deben llevar a cabo un conjunto de acciones y procedimientos, con el propósito de obtener la información mencionada anteriormente; en consecuencia, se pueden realizar estudios sobre los orígenes y causas del desastre a través de revisiones históricas de la zona, realizando estudios geológicos y antropológicos, junto con percepciones que tenga la propia comunidad sobre las amenazas existentes y estudios sobre la vulnerabilidad de la

infraestructura presente, realizar un seguimiento constante sobre la amenaza, entre otra gran variedad de procedimientos destinados a la obtención de dicha información [11] [7].

A su vez, en esta etapa, es donde se diseñan y se especifican medidas y acciones a tomar con el fin de impedir o reducir la probabilidad de ocurrencia de fenómenos que puedan causar un desastre [5].

Un ejemplo claro de prevención puede ser la reubicación de viviendas que se encuentran en zonas inundables (escenario muy común a veces causado por un alto crecimiento poblacional), situación sumamente compleja para comunidades consolidadas y con un gran costo desde el punto de vista económico.

Por lo tanto, resulta claro que en temas de prevención, son necesarios en algunos casos programas a largo plazo, de manera de lograr desarrollar la comunidad considerando factores potencialmente dañinos [6]. Por ejemplo, para evitar el caso planteado anteriormente, debería implementarse un plan para el desarrollo urbano de la región, indicando localizaciones de industrias y realizando un correcto ordenamiento urbano y territorial.

En conclusión, a partir de los datos obtenidos se logra identificar qué zonas son las más propicias para la localización o reubicación de asentamientos, construcción de infraestructura, etc., de modo que de producirse un desastre, el impacto sea el menor posible [7]. Esta etapa, es de gran importancia para la etapa de mitigación, pues necesita conocer los datos aquí relevados [11].

1.3.1.2 MITIGACIÓN

El propósito de la mitigación, es intervenir para reducir/modificar las características de una amenaza y/o reducir las vulnerabilidades existentes en una comunidad, con el fin de lograr disminuir los efectos dañinos de un potencial desastre, y en consecuencia reducir el riesgo que enfrenta dicha comunidad ante la ocurrencia del mismo [7] [6].

La etapa de mitigación es muy importante en la gestión de riesgos, porque si bien las medidas de prevención buscan reducir las probabilidades de ocurrencia de un desastre en la comunidad, no siempre es posible garantizar que este no ocurrirá (o los costos de ejecutar los planes de prevención son demasiado elevados), de esta manera, aplicando los planes llevados a cabo en la etapa de mitigación, se puede lograr disminuir el impacto del mismo sobre la comunidad [7].

1.3.1.3 PREPARACIÓN

En la etapa de preparación, se busca organizar, estructurar y planificar la respuesta ante la ocurrencia de un desastre [7].

Con tal fin, se realizan planes operativos de emergencia, donde se planifican las

acciones a tomar en temas tales como los de búsqueda, salvamento, socorro y asistencia entre otros, de manera de que si ocurre un desastre se pueda actuar inmediatamente siguiendo un plan eficiente y específico, elaborado y estudiado con anterioridad, de forma de optimizar al máximo el tiempo y los recursos, ya que las primeras horas luego de ocurrido un desastre son factores críticos.

También, es necesario elaborar de antemano, un detallado inventario sobre los recursos disponibles, así como establecer los estados de alerta y las acciones que la población y las organizaciones deben tomar ante la declaración de dichos estados [7] [6].

1.3.1.4 ALERTA

El estado de alerta sirve para prevenir a la comunidad ante la ocurrencia de un desastre inminente, de manera de poder tomar acciones específicas en pro de reducir potenciales daños [7].

Una de las complicaciones mayores para poder brindar una alerta, es que muchas veces, no es posible percibir la ocurrencia de un desastre hasta apenas minutos antes del mismo, lo cual hace imposible declarar un estado de alerta.

De esta manera, dependiendo de la probabilidad de acierto de la estimación del desastre y la gravedad prevista del mismo es que se establecen los diferentes estados de alerta [7].

Generalmente, los estados de alerta se especifican mediante colores, de manera de que sea fácilmente interpretable por la población; los mismo son difundidos a través de los medios de comunicación [7] [6].

1.3.2 FASE POSTERIOR A LA OCURRENCIA DE UN DESASTRE

Una vez ocurrido un desastre, resulta de gran importancia brindar ayuda a las zonas afectadas, de manera de lograr cubrir las necesidades básicas, y realizarlo tan rápido como sea posible.

Por otra parte, también es necesario lograr llevar a la comunidad a un estado de desarrollo similar al momento pre-desastre o mejor [9]. Para lograr lo mencionado anteriormente, es necesario aplicar una serie de procedimientos y acciones que requieren gran coordinación y esfuerzo.

1.3.2.1 RESPUESTA

Esta etapa se ejecuta inmediatamente ocurrido el desastre; aquí se desarrollan las actividades de búsqueda, rescate, acciones de socorro, evacuación de afectados, alojamiento temporal y distribución de suministros. Todas estas acciones, buscan reducir el impacto del desastre, buscando salvar vidas y reducir el sufrimiento de

las personas afectadas [8] [6].

Otros objetivos de esta etapa, son disminuir el daño sobre las propiedades, preservar el medio ambiente y resguardar la estructura social, económica y política de la comunidad [8].

1.3.2.2 MANEJO DE SUMINISTROS

El manejo de suministros en la etapa de respuesta es un punto crítico y si no es manejado adecuadamente puede volverse caótico, resultando en gastos realmente elevados referentes a la logística [6].

Esto sucede porque muchas veces (generalmente cuando el desastre es de grandes dimensiones y toma gran difusión pública) se reciben suministros (donativos) nacionales e internacionales de gran variedad, como ser medicamentos, alimentos, vestuario, maquinaria, etc., por lo que resulta sumamente complejo, en base a las necesidades inmediatas, lograr distribuir de manera eficiente los suministros recibidos. A su vez, muchas veces, existen donativos que no han sido solicitados, y que su utilidad es de dudosa práctica, debiendo utilizar recursos en gestionarlos y hasta en algunos casos provocando obstáculos a la llegada de suministros vitales (se congestionan los medios de recepción, por ejemplo aeropuertos); todo esto complica aún más el panorama existente [14].

Considerando los problemas planteados anteriormente, en 1992 la OPS (Organización Panamericana de la Salud) desarrolló una herramienta de software llamada SUMA, destinada a la organización y manipulación de suministros [22]. En el Anexo I se estudia en profundidad el software SUMA, brindando un detallado estudio de sus características y funcionalidades.

Algunas medidas a tomar desde el punto de vista del manejo de suministros antes de que ocurra el desastre pueden ser: identificar lugares estratégicos de depósito, tomar medidas a nivel gubernamental en términos de aduana para acelerar la recepción de donaciones del exterior, capacitar personal para el adecuado manejo de suministros (capacitarlos en el uso de software de apoyo si se utiliza, por ejemplo SUMA), entre otras [14].

1.3.2.3 REHABILITACIÓN

Es la etapa de restitución (a corto plazo) de algunos servicios básicos de la comunidad, como ser la recuperación de los servicios de salud, programas educativos, energía, caminos, medios de comunicación, agua, desagüe y demás servicios indispensables para el correcto funcionamiento de una sociedad [5] [7].

Cabe destacar, que durante la etapa de rehabilitación, no cesan las actividades referentes a la etapa de respuesta, sino que se continúa con las mismas en paralelo, y a medida dentro de lo posible, se comienzan a desarrollar las actividades correspondientes a la Reconstrucción [5].

1.3.2.4 RECONSTRUCCIÓN

En la etapa de reconstrucción, se llevan a cabo los planes correspondientes a la recuperación de la sociedad, ya que en ocasiones, la ocurrencia de un desastre deja a las comunidades con grandes daños a nivel físico, económico y social [5] [7]. De esta manera, son necesarios llevar adelante procesos de mediano y largo plazo capaces de reflotar la economía de la región, creando nuevas fuentes de trabajo y recuperando industrias deterioradas, reparar las infraestructuras dañadas principalmente a nivel de viviendas, trabajar a nivel social para tratar a las personas que hayan sufrido pérdidas familiares o puedan haber quedado con secuelas traumáticas luego del desastre, etc. El objetivo ideal de esta etapa es que al finalizarla la comunidad se encuentre en un nivel de desarrollo igual o superior al que se encontraba antes de ocurrido el desastre [5] [6].

Vale aclarar, que durante la reconstrucción de la comunidad, deben de considerarse, y hacerse cumplir en la mayor medida posible, las medidas provenientes de los planes de prevención y mitigación, a la vez de aplicar en la medida de lo posible las “lecciones aprendidas” dejadas por el desastre, con el fin de reducir el impacto de un nuevo desastre [5].

1.4 SINAE

El SINAE (Sistema Nacional de Emergencias), creado por el Estado Uruguayo en la ley N°18621 como un sistema público de carácter permanente, tiene por objetivo (redactado en dicha ley) la “protección de las personas, los bienes de significación y el medio ambiente, ante el acaecimiento eventual o real de situaciones de desastre, mediante la coordinación conjunta del estado con el adecuado uso de los recursos públicos y privados disponibles, de modo de propiciar las condiciones para el desarrollo nacional sostenible” [94].

Es responsable de realizar la coordinación de los recursos y acciones a llevarse a cabo por los distintos organismos, ya sean públicos, privados u organizaciones sociales, para efectuar las distintas acciones que forman parte de cada una de las etapas de la gestión de riesgos (prevención, mitigación, rehabilitación, etc.), haciendo un uso eficiente de los recursos (humanos, técnicos, económicos, etc.).

Se encuentra integrado por un conjunto de organismos, donde cada uno tiene un rol y responsabilidades específicas dentro del sistema, existiendo a su vez una coordinación entre ellos. Dichos organismos que forman el Sistema Nacional de Emergencias son el Poder Ejecutivo, la Dirección Nacional de Emergencias, la Comisión Asesora Nacional para Reducción de Riesgo y Atención de Desastres, los Ministerios, entes autónomos y servicios descentralizados, los Comités Departamentales de Emergencias y los Centros Coordinadores de Emergencias Departamentales [94].

Finalmente, es importante mencionar que según este organismo, los principales

desastres que afectan al país y causan mayores daños son las inundaciones, seguidos por los incendios forestales y las tormentas de viento [95].

2 LOGÍSTICA

2.1 QUE ES LOGÍSTICA

Cuando se habla de logística en general, se hace referencia a la organización, traslado y almacenamiento de cosas o personas. La tarea principal es distribuir los productos adecuados en buenas condiciones (tiempo y cantidad) a ciertos lugares determinados [35].

Tradicionalmente, las decisiones logísticas son clasificadas como estratégicas, tácticas y de operaciones [80]. Las decisiones estratégicas tienen efectos a largo plazo, incluyen decisiones de diseño y adquisición de recursos importantes. Las tácticas incluyen decisiones de planificación de producción, planificación de distribución y selección de medios de transporte, en general aplican en el mediano plazo. Las decisiones operativas se hacen en tiempo real, se basan habitualmente en datos muy detallados y tienen un alcance limitado..

Una cadena de suministro es un sistema de logística complejo en el que las materias primas se convierten en productos terminados y luego son distribuidos a los usuarios finales (consumidores o empresas). Incluye proveedores, centros de producción, almacenes, centros de distribución y puntos de venta [80].

2.2 LOGÍSTICA HUMANITARIA

Se vincula a la logística humanitaria, a las actividades logísticas que tienen lugar en el marco específico de la acción humanitaria de emergencia. De forma más concreta, en los desastres repentinos que requieren una rápida movilización de recursos materiales y humanos para atender a las víctimas de la emergencia.

Principalmente se tiene como objetivo atender las necesidades de las personas afectadas por una situación de emergencia o desastre y minimizar o suavizar el sufrimiento de las víctimas. Para alcanzar este objetivo, puede ser muy productivo aplicar el concepto de logística, por ejemplo, en las decisiones de distribución de bienes y servicios necesarios para una situación de emergencia. Esto es un aspecto fundamental a destacar en cuanto a objetivos de la logística humanitaria.

Si se utilizan los principios logísticos, las acciones y operaciones dentro del periodo de la emergencia, pueden ser más rápidas, ágiles y efectivas. Se habla de utilizar los principios logísticos en el sentido de las tareas de movilización de personas, distribución de material, localización de refugios y más acciones necesarias para dar asistencia a los afectados y lograr implementar todas las acciones de respuesta [3].

Dentro de la logística humanitaria resulta de interés mencionar distintas disciplinas además de las ya señaladas. La gestión de inventarios y almacenamiento de suministros están fuertemente relacionadas (al igual que en muchos sistemas logísticos en general). También debe considerarse la planificación de evacuaciones, la localización de refugios o albergues temporales, el transporte de personas capacitadas para colaborar y la distribución de insumos necesarios como actividades importantes dentro de la logística humanitaria.

A continuación se describe brevemente el modelo de [1]. Se pueden identificar fases en la ayuda humanitaria de emergencia: mitigación, preparación, respuesta y rehabilitación, conceptos ya abordados que en esta ocasión se describen desde el punto de vista de las acciones de cada fase.

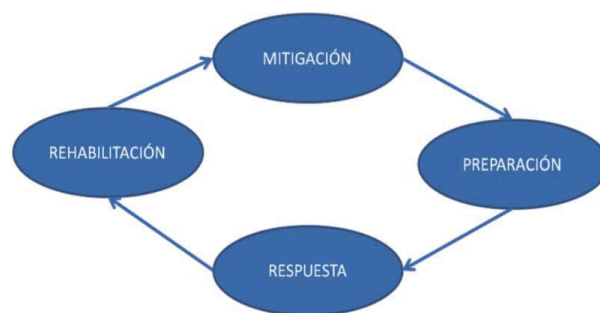


Figura 2 – Fases de emergencia [1]

Mitigación se llama a la toma de medidas preventivas que ayuden a disminuir la probabilidad de ocurrencia de un desastre o minimizar los daños en el caso de que ocurra [1].

Una buena gestión se basa en prever e identificar los problemas que puedan surgir y proveer los suministros específicos.

La fase de preparación refiere a la planificación de las acciones que tendrían lugar en caso de producirse un desastre. No es deseable que las actividades logísticas se improvisen en el momento de ocurrido un desastre. Debería existir una planificación y preparación previa ya que muchas de las necesidades que los desastres generan pueden ser previsibles. Algunos aspectos a considerar en la elaboración de este plan son qué tareas se deben realizar, quiénes son responsables de ellas, qué recursos son necesarios, cómo se consiguen, así como también qué acciones se deberán tomar en caso de imprevistos. La planificación debe estar basada en un profundo conocimiento geográfico, social y político de las zonas en cuestión [3].

La fase de respuesta toma protagonismo durante e inmediatamente después del desastre. Algunas de las tareas que abarca son la asistencia, movilización de suministros, evacuación y reubicación de poblaciones afectadas. Puede existir un sistema logístico para que las acciones de respuesta sean rápidas, ágiles y efectivas [1].

Hay una enorme carga logística en la fase de respuesta asociada a la cadena logística de suministros, que se trata de la entrega en los lugares y en el tiempo apropiado de las provisiones, en condiciones y en las cantidades requeridas, abarcando las actividades de abastecimiento, transporte, almacenamiento y distribución. Las actividades requeridas para el despliegue logístico deberían ser preparadas, probadas y conocidas por todos los actores [1].

Rehabilitación es la fase que busca llevar a la comunidad a un estado similar al previo de la ocurrencia del desastre, tratando de mejorar las condiciones de la población afectada; generalmente esta fase requiere de planes a largo plazo [1].

Ante el acontecimiento de un desastre natural, es importante considerar el probable daño en infraestructuras, limitando significativamente la capacidad de transporte en las zonas afectadas. De esta manera, surgen nuevas restricciones, como el bloqueo de rutas y disminución o destrucción de vías de comunicación, afectando planes y dificultando las actividades de respuesta [35].

Como se mencionó anteriormente, una de las herramientas más conocidas referentes a la logística humanitaria es el software SUMA [punto 8.2], siendo de gran ayuda principalmente en la gestión y control de inventarios [22]. Sin embargo, en la actualidad el SUMA ha caído en desuso, dando cabida al software LSS (Logistics Support System). El mismo, desarrollado como sucesor del SUMA, mejora la coordinación a nivel nacional e internacional entre todos los agentes humanitarios, brindando un mejor y eficiente intercambio de información entre los mismos (ofrece la posibilidad de trabajar en dos plataformas, una de escritorio y otra web). Esto conlleva a un mejor manejo de suministros y una más rápida respuesta a las necesidades de la población afectada [87].

Otro software muy utilizado en el contexto de la logística humanitaria es HELIOS (desarrollado por el instituto Fritz como sucesor del software HLS –Humanitarian Logistic Sistem). El mismo es una herramienta que se utiliza para coordinar y gestionar los suministros, ayudando así en la toma de decisiones. Permite realizar el seguimiento de la cadena de suministros, pudiéndose especificar el envío o recepción de los suministros, así como también generar reportes informativos [13]. A su vez, al ejecutarse sobre la plataforma web, se favorece el rápido intercambio de información entre los distintos agentes. Todo esto, repercute en tiempos de respuestas mejores y un manejo más efectivo de los suministros, así como también en una disminución de costos y esfuerzos en materia de gestión de suministros.

Finalmente, otra herramienta de gran utilidad en el contexto de la logística humanitaria es el software Sahana Eden, desarrollado por la Fundación Sahana. Es una herramienta flexible que puede ser utilizada en varias de las fases de la gestión de riesgos, tales como mitigación, preparación, respuesta y reconstrucción. A su vez, es un software libre y de código abierto (bajo la licencia LGPL), basado en la plataforma web, y cuenta con un conjunto de módulos que proveen soluciones a diferentes problemas relacionados con la gestión de riesgos [83].

Algunas de las funcionalidades principales del software son el registro de personas extraviadas, administración y coordinación de suministros (proporcionando

funcionalidades como solicitar/comprometer suministros, manejo de inventarios y catálogos), registro de albergues de emergencia y coordinación de voluntarios [83].

2.3 DIFERENCIAS ENTRE LOGÍSTICA HUMANITARIA Y LOGÍSTICA COMERCIAL

El primer aspecto a diferenciar, es el objetivo detrás de cada una de las actividades. Mientras que en la logística comercial se busca maximizar beneficios (por ejemplo minimización de costos), en la humanitaria el objetivo que se persigue es el de atender de manera efectiva las poblaciones afectadas (aunque se busca eficiencia, la minimización de costos no es el principal objetivo) [1].

Las actividades de acción humanitaria se caracterizan por ser de corto plazo, ya que se busca suministrar insumos necesarios de manera rápida, ya sean productos o servicios, a las personas afectadas por un desastre. Teniendo en cuenta que el factor tiempo es muy limitado en estas situaciones, la clave en el desarrollo de la logística en las acciones humanitarias es asegurar una respuesta rápida.

Otra diferencia importante entre los dos tipos de cadenas es el patrón de demanda. Para muchas cadenas de suministro comercial, la demanda externa de los productos es relativamente estable, conocida y desde lugares fijos. Generalmente para la cadena comercial, las demandas se producen desde ubicaciones establecidas en intervalos relativamente regulares. Por otra parte, las demandas de la cadena de suministros en el contexto de logística humanitaria son irregulares, en intervalos de tiempo irregulares y ocurren repentinamente. Lo que se demanda, son suministros y personas [86].

Dentro de la logística comercial se utilizan métodos bien definidos para determinar los niveles de inventario en función del tiempo de espera, la demanda y los niveles de servicio al cliente de destino. Por su parte, en la logística humanitaria estos factores son difíciles de controlar por las grandes variaciones en los plazos de entrega, las demandas y las ubicaciones de demanda [86].

La infraestructura y las personas que forman parte de una cadena de suministro comercial son permanentes en el tiempo y existen métodos bien definidos para determinar la localización de centros de distribución [86]. En el caso del suministro humanitario es completamente temporal, la infraestructura logística se implementa inmediatamente después de ocurrido el desastre. Una vez finalizada la acción humanitaria, el lugar afectado debería volver a la normalidad.

Cuando ocurre una situación de emergencia, organizaciones de otros lugares suelen realizar todo tipo de donaciones para ayudar a los afectados, esto puede llevar muchas veces ocurran donaciones no solicitadas, lo que conlleva a que las

donaciones no se corresponden con las necesidades reales de la población afectada [4]. Esto puede provocar problemas debido a una carga extra de recursos que quizás hasta sean innecesarios, aumentando el volumen de trabajo operativo, causando por ejemplo problemas de espacio en depósitos y puertos [1]. Esto significa que se asignen menos recursos para productos más importantes que son realmente necesarios para distribuir, empeorando la efectividad de la gestión.

En cuanto a los aspectos económicos en el contexto de logística humanitaria, se puede resaltar la dificultad para obtener fondos para las fases de mitigación, preparación y rehabilitación ya que es más fácil la recaudación de fondos para atender las víctimas enseguida se manifiesta un desastre y se reciben donaciones o apoyo externo de manera inmediata. La efectividad de la logística depende fuertemente del nivel de inversión en lo que refiere a transporte y planificación en la fase de prevención [1].

3 GESTIÓN DE INVENTARIOS

3.1 INTRODUCCIÓN

La gestión de inventarios es un problema clave en la planificación y operaciones de sistemas logísticos. Inventarios son reservas de ítems (bienes materiales) esperando para ser procesados, transportados o usados en un punto de la cadena de suministros.

El objetivo de la gestión de inventarios es la reducción de costos de logística porque sirve para poder satisfacer las necesidades a veces aleatorias de demanda y plazos de entrega. Un aspecto muy beneficioso de mantener inventarios, es el desacoplamiento de las distintas componentes de una línea de producción, esto quiere decir, que si ciertos sectores trabajan más lentos que otros, al tener reservas de materiales, no sea necesario parar el proceso de producción [79]. Permite también que ciertos productos que, por alguna razón no se pueden obtener todo el año, como frutos de estación, puedan estar disponibles todo el año.

La planificación y el control del volumen de flujos de los ítems, son claves en las que se basa la gestión de inventarios [79].

En la gestión de inventarios existen modelos de reabastecimiento de inventario que tratan de reducir costos, de esta manera, con dichos modelos se puede saber en qué momento se debe realizar una orden y qué volumen de materiales se debe pedir, en [85] se desarrolla un modelo muy interesante.

Es intuitivo concluir, que si se reabastece el inventario en periodos cortos de tiempo, la cantidad pedida debe ser pequeña, lo cual reduce el costo de almacenamiento pero se incrementa la cantidad de pedidos (aumentando el costo monetario y logístico); si se repone el inventario en periodos largos de tiempo, la cantidad pedida debe ser grande lo cual reduce el costo de hacer el pedido pero incrementa el costo de almacenamiento.

Por lo tanto, la gestión de inventarios ayuda a determinar los niveles de stock que permiten minimizar costos operativos y a su vez, satisfacer los requerimientos de la demanda. Esto involucra la selección de qué productos se mantienen almacenados (en stock), decidir qué niveles de inventario mantener y elegir las posibles ubicaciones de los establecimientos de almacenamiento.

3.2 APLICACIÓN AL CASO DE ESTUDIO

Muchas veces, cuando ocurre un desastre natural, las personas afectadas que se

ven en situaciones extremas, suelen ser evacuadas durante la emergencia. En muchos casos luego de la emergencia sus viviendas han sido destrozadas y no tienen hacia donde volver. Por eso es pertinente que la distribución de los bienes y servicios necesarios para aliviar el sufrimiento y sobrellevar la situación de la mejor manera posible sean suministrados tanto durante como luego de ocurrida la emergencia. Cuando se hace referencia a la distribución de determinada cantidad de suministros por un periodo de tiempo, se hace alusión a la toma de decisiones operativas en general y es igualmente útil para el caso de la logística humanitaria en particular.

3.3 GESTIÓN DE INVENTARIOS: RELEVAMIENTO DE COSTOS

Al mismo tiempo, mantener niveles de stock puede resultar costoso debido a que representa una gran cantidad de capital estancado. Los costos relevados para determinar una política de inventarios clásica se pueden clasificar en cuatro categorías [80]:

- Costos de obtención (asociados a la adquisición de bienes)
- Costos de tener un inventario (cuando los bienes son guardados por un periodo de tiempo, hay que considerar la diferencia monetaria entre, el costo de tener bienes guardados y el interés generado si se hubiera invertido el dinero (interés bancario por ejemplo). También hay que considerar los costos de energía, espacio, sueldos de personas, etc.
- Costos de escasez (en nuestro caso de estudio, estos costos dependerán de las necesidades variables de las personas afectadas)
- Costos de obsolescencia (los bienes guardados pueden perder valor con el transcurso del tiempo, por ejemplo alimentos perecederos)

3.4 MODELOS DE GESTIÓN DE INVENTARIOS

Existen múltiples modelos y algoritmos representando distintos casos de realidad y tratando de resolver las distintas necesidades que plantea la gestión de inventarios; en esta sección se presentan algunos modelos planteados por Ghiani et al. [80].

3.4.1 GENERALIDADES

Se dice que un modelo es determinista, cuando se toma por supuesto que las

demandas, los precios y los plazos, no varían significativamente y se conocen de antemano [80].

Un modelo es estocástico, cuando algunos datos son inciertos, agregando alta complejidad e inexactitud, concluyendo que muchas veces no es posible satisfacer toda la demanda [80]. Por lo tanto, es de suponer que el modelo estocástico es el que mejor se adecua a las necesidades de la logística humanitaria, dado que el escenario es bastante caótico y hasta impredecible. La demanda de suministros varía en tiempo real y es muy difícil de determinar.

Muchas veces, por simplicidad se puede derivar a un modelo con un único punto de almacenamiento, los modelos con múltiples puntos son generalmente mucho más complejos.

Los problemas de los modelos de gestión de inventarios, que gestionan un número grande de ítems distintos y múltiples puntos geográficos de almacenamiento generalmente son clasificados como NP-hard, aunque se acercan bastante más a la realidad que se pretende analizar.

3.4.2 ÚNICO PUNTO DE ALMACENAMIENTO, UN SOLO ÍTEM Y DEMANDA CONSTANTE

Este modelo [80] no maneja el problema de la demanda incierta pero es un modelo simple que puede ser aplicado a la realidad.

- tasa constante de la demanda,
- tiempo transcurrido entre pedidos,
- el tamaño del pedido (i.e. la cantidad de producto pedida en cada reabastecimiento).

Las cantidades son tales que:

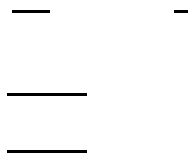
- el valor de un ítem.
- el costo de realizar un pedido.
- el costo de guardar un ítem, por unidad de tiempo.
- el costo de almacenamiento, independiente del tiempo en el que se encuentre almacenado.
- el costo de almacenamiento, por unidad de tiempo.

Se puede expresar:

El parámetro p es la tasa de interés bancario (mide el costo de capital).

El problema consiste en determinar q para que el costo medio total por unidad de tiempo sea mínimo.

Para reabastecimiento instantáneo (esto es, el inventario se reabastece instantáneamente cuando llega a cero) con la llegada del lote pedido, se pueden deducir y

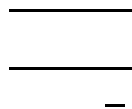


Se puede deducir que el tiempo de ciclo de pedido es

—

Este modelo, con estos supuestos, coincide con el modelo clásico EOQ (Economic Order Quantity)

Para una política de reabastecimiento no instantáneo según [80] se puede expresar de la siguiente forma:



Sobre el modelo EOQ

Es muy popular y utilizado en muchas empresas a nivel mundial. Se caracteriza por su sencillez a la hora de calcular la cantidad por orden o pedido. Así mismo, los supuestos que introduce este modelo facilitan su aplicación pues se asume la existencia de variables constantes como la demanda [81].

El hecho de que la demanda sea constante se aleja de la realidad, donde se encuentran demandas estacionales, demandas irregulares. La demanda será uno de los elementos más inestables a los que se enfrentará a la hora de planificar su producción. En algunos casos, esta incertidumbre a la hora de predecir la demanda puede forzar la utilización de métodos probabilísticos para facilitar el cálculo de la cantidad óptima por pedido.

Este método considera que el nivel de inventario se reabastece instantáneamente, es decir, al instante en que se agota un ítem, se abastece. Es una hipótesis que la práctica no va a ocurrir en la mayoría de los casos.

En [85] se plantean modelos y deducciones un poco más interesantes y más aplicables al contexto de la logística humanitaria. Se plantean las cuestiones principales a considerar durante y después de que ocurre un desastre:

- Establecer centros de almacenamiento de suministros necesarios,

- recepción y distribución de los mismos.
- Instalar un sistema de gestión de inventario que administre y distribuya suministros a los evacuados.
- Poner en práctica un innovador proceso de rotación del stock, que asegure que suministros costosos no expirarán en el almacenamiento.

El objetivo del modelo planteado en [85] es encontrar la cantidad de stock tal que el proceso de distribución sea ininterrumpido con probabilidad 1-e y a costo mínimo. Se utilizan conceptos y deducciones que nacen de la programación estocástica, cuyo objetivo, como en nuestro caso, es encontrar una decisión óptima para problemas en los que se involucran datos inciertos.

3.4.3 MODELO CON MÚLTIPLES ÍTEMS

Este caso es un poco más aplicable a la realidad de la logística humanitaria en general. Se usa la misma nomenclatura que el modelo anteriormente desarrollado. Ahora pueden existir muchos ítems, el objetivo es el mismo, encontrar la cantidad de producto pedida en un reabastecimiento.

Para trabajar con el supuesto de múltiples ítems, se agrega complejidad. Por ejemplo, se agregan algunas restricciones como la capacidad del espacio de almacenamiento, así como también restricciones asociadas a los costos de realizar pedidos de abastecimiento. Se describe el modelo de [86] que se desarrolla bajo las hipótesis del EOQ.

Se puede formular el problema de la siguiente manera, con una función objetivo a minimizar que representa el costo total promedio por unidad de tiempo, sujeto a las restricciones que aplican

Usando las hipótesis de EOQ (niveles de demanda determinados) se puede afirmar que

Usando la ecuación se llega a la siguiente igualdad

$$\text{---} \quad \text{---}$$

Si consideramos que las restricciones de capacidad son lineales, se puede expresar

Donde a y b son constantes. A continuación se presenta una simple heurística de dos pasos que se puede utilizar si se considera que las tasas de interés son idénticas para todos los ítems i.e.

Paso 1 – calcular λ_i con la formula siguiente

$$\lambda_i = \frac{a}{b + \sum_{j=1}^n x_j}$$

Si se satisfacen las restricciones de capacidad parar

Paso 2 – Aumentar la tasa de interés p una cantidad Δp a determinar, entonces:

$$\lambda_i = \frac{a}{b + \sum_{j=1}^n x_j}$$

Se determina el valor de λ_i para que se cumpla la igualdad

Entonces

$$\lambda_i = \frac{a}{b + \sum_{j=1}^n x_j}$$

Luego, insertando el valor de λ_i en λ_i , queda determinado el valor de λ_i .

4 DISTRIBUCIÓN DE SUMINISTROS

4.1 INTRODUCCIÓN

En muchas ocasiones, las catástrofes golpean a los países de forma repentina (principalmente las catástrofes naturales), y muchas veces provocando un gran impacto en las zonas afectadas y a la sociedad en general. Ante tal suceso, y principalmente si la catástrofe es de gran magnitud (o muchas veces si se produce la suficiente “publicidad” del hecho por parte de los medios de comunicación), no es de extrañar que el país afectado reciba gran cantidad de ayuda y colaboración internacional (países, organizaciones, empresas, particulares, etc.) [35].

A fin de cuentas, se contará con una gran cantidad de mercadería (generalmente muy variada) que deberá ser manejada y distribuida de la forma más rápida y eficiente posible, de modo de lograr cubrir de la mejor manera las necesidades de las personas en las zonas afectadas.

Por otra parte, la demanda existente, puede variar de un segundo a otro, por ejemplo en el caso de terremotos, una zona que puede no estar necesitando una gran demanda de suministros, tras sufrir una réplica, pueden causarse estragos que hagan que su demanda cambie significativamente. Por lo tanto, también es necesario contar con una rápida adaptación al cambio, tanto de la demanda como de las zonas demandantes [1].

Sumado a lo anterior, en muchos casos puede no contarse con la infraestructura necesaria como para llevar a cabo la distribución de suministros (escases de vehículos, rutas intransitables, aeropuertos inutilizables, etc.) ni con infraestructura donde alojar y organizar, para luego poder distribuir dichos suministros (edificios dañados o ausencia de los mismos, falta de personal calificado, etc.).

Para hacerle frente a los problemas planteados anteriormente, entre otros muchos más, y poder resolverlos de forma eficiente, se utiliza la logística humanitaria, en donde, ante un panorama donde se conocen las limitaciones existentes, se intenta realizar una distribución de suministros que logre minimizar el sufrimiento y satisfacer en la mayor medida posible, la demanda de las personas afectadas.

Tricoire et al. [16] analizan a cada pueblo afectado como un potencial Centro de Distribución, decidiendo finalmente cuáles de ellos los serán.

Para esto, consideran que las personas residentes en cada uno de los pueblos concurrirán al Centro de Distribución que tengan más cercano (se especifican valores lógicos dentro de los cuales una persona puede ir caminando en busca de sus suministros), y de esta manera se busca cubrir en la mayor medida posible la demanda de cada uno de los pueblos. A su vez, cada centro de distribución tiene un costo de apertura y mantenimiento (dependiendo del pueblo), por lo que también

se busca reducir (en lo posible) estos costos.

En [14] se plantea un modelo de optimización multi-criterio considerándose factores como tiempo y costo en la distribución (entre otros). A su vez, también definen factores de compatibilidad entre los vehículos y los “arcos” (definidos por los autores como el camino entre dos pares de nodos), pudiéndose especificar, por ejemplo, que vehículos de gran tamaño no pueden transitar por caminos angostos. También se especifican velocidades para los arcos y los vehículos, definiéndose una relación entre la velocidad con la que circula un vehículo a través de un arco, la velocidad permitida en el arco y la velocidad máxima del vehículo (en caso de que viajen varios vehículos juntos -convoy- la velocidad de todos ellos debe ser la misma velocidad a la que circula el vehículo más lento).

Por su parte, Bozorgi-Amiri et al. [17] plantean un modelo para la distribución de suministros (desde los proveedores, a los centros de distribución y finalmente a las zonas afectadas) pero toman en cuenta la existencia de factores de incertidumbre, planteando distintos escenarios junto con las probabilidades de que estos ocurran; esto modifica variables como el costo de la distribución, posibilidad de distribuir los suministros a determinados centros de distribución o zonas afectadas, costos de apertura de algunos centros de distribución, etc.

Knott et al. [18] desarrollan un modelo para optimizar el volumen de carga de los camiones, de manera de reducir costos en la cantidad de camiones necesarios y maximizar la cantidad de suministros transportados.

Angelis et al. [19] proponen un modelo para la distribución de suministros utilizando aviones, cuyo objetivo principal es maximizar la demanda satisfecha, sin considerar demasiado los costos. Se consideran varios depósitos, hangares y aviones; fue desarrollado pensando en el Programa Mundial de Alimentos (WFP) en Angola.

Por su parte, Ortuño et al. [15] plantean un modelo que se enfoca en un conjunto de problemas comunes existentes a la hora de planificar y realizar la distribución de suministros en una zona afectada, generalmente en condiciones adversas. Para esto, se utiliza una metodología multi-criterio, de forma de poder considerar varios factores a la hora de tomar la mejor decisión logística, esto se lleva a cabo utilizando la programación por metas, donde a su vez existen niveles de prioridades entre las diferentes metas; en el Anexo II puede observarse una breve explicación de en qué consiste la programación por metas.

Desde el punto de vista de la logística humanitaria, uno de los principales objetivos luego de que ha ocurrido un desastre, es asegurar que los suministros existentes lleguen tan rápido como sea posible a las zonas afectadas de forma de satisfacer en la mayor medida posible la demanda, utilizando de manera eficiente las rutas existentes [1].

A su vez, es de suponer que existan diferentes tipos de vehículos con diferentes características (velocidad, capacidad de carga, costo de mantenimiento, etc.) que puedan servir para el reparto de los suministros, y también se contará con un

presupuesto (por lo general ajustado) para cubrir los gastos que genere la operación.

En la sección siguiente (punto 4.2) se desarrollara el último modelo mencionado [15], pues consideramos que es un modelo muy completo, que toma en cuenta muchas de las condiciones necesarias a atender en la distribución de suministros en un contexto de ayuda humanitaria, y a su vez logra un nivel de abstracción adecuado de manera de ser completo pero al mismo tiempo no extremadamente complejo (lo cual conllevaría a un modelo que simule fielmente la realidad pero que se torne poco aplicable debido a su complejidad y por ende difícil resolución).

4.2 DESCRIPCIÓN DEL MODELO

Como se mencionó en la sección anterior, a continuación se procede a explicar el modelo desarrollado por Ortuño et al. [15].

Cabe mencionar que este modelo se utiliza como modelo base del sistema “Humanitarian Aid Distribution System” (HADS). Este sistema, desarrollado por los mismos autores que plantean el modelo, busca brindar apoyo al momento de planificar y llevar a cabo la distribución de suministros en un ambiente donde ha ocurrido un desastre [15].

Es importante aclarar que el sistema HADS actualmente se encuentra en etapa de desarrollo, contándose en la actualidad con un pequeño prototipo (desarrollado en 2010), no estable y con funcionalidades acotadas, siendo de escasa aplicabilidad a un caso real; a su vez, para calcular la solución se busca la solución óptima del modelo, lo cual puede llevar a tiempos de procesamiento demasiado elevados para el contexto en el cual se está trabajando (donde generalmente se requieren soluciones rápidas) [88].

En el modelo se define un Mapa Logístico donde se representan los nodos a considerar y las conexiones existentes entre ellos. Básicamente se distinguen entre tres tipos de nodos, los demandantes, los nodos proveedores y los nodos intermedios. El objetivo de esta distinción es conferirles determinados atributos de interés a los distintos tipos de nodos, por ejemplo, en los nodos proveedores se puede especificar la cantidad de suministros existentes y en los demandantes la cantidad de suministros que se están necesitando. En los nodos intermedios, por su parte, otros nodos podrán depositar suministros para que queden cerca de los nodos “demandantes” (así cuando estos tengan nueva demanda se los pueda abastecer más rápidamente).

Cabe aclarar, que a los efectos prácticos, los suministros en el modelo se trabajan como “unidades”. Esto significa que por ejemplo, en el modelo una unidad de suministro se puede corresponder con un paquete que contengan elementos

básicos necesarios, de esta manera, el modelo no distingue entre distintos tipos de suministros sino que trabaja siempre con un solo tipo de suministro (el usuario es el que define qué correspondería a una unidad de suministro).

En el caso de la demanda de suministros en cada nodo, este valor es un objetivo que se intenta alcanzar, pero que muchas veces no llega satisfacerse del todo, esto puede deberse, por ejemplo, a que la cantidad de suministros existentes a distribuirse no alcance para satisfacer a cada nodo, buscándose una solución lo más equitativa posible.

También se pueden definir distintos tipos de vehículos, especificando la cantidad de cada uno de ellos en cada nodo. Además, de cada vehículo, es de interés saber su velocidad promedio, capacidad y costos de mantenimiento y consumo (entre otros detalles) para poder considerar todos los factores a la hora de decidir que vehículo enviar a determinado nodo.

De manera similar a como ocurre en los nodos y los vehículos, en las conexiones entre los nodos (rutas), también se pueden definir determinados atributos de interés como ser la distancia de esa ruta, la velocidad que se puede desarrollar, probabilidad de saqueo, etc.

Vale aclarar que cada camión se encuentra asociado a un nodo particular, y la manera en que operan los camiones en el sistema es la siguiente:

Si un camión asociado al nodo A lleva suministros al nodo B, luego de dejar los suministros en el nodo B, el camión vuelve al nodo A por el mismo camino por el que fue de A a B (por lo que para calcular el costo de transportar suministros de A a B se debe considerar también el costo de transportar los camiones vacíos de B a A –el costo de transportar un camión vacío de A hacia B es el mismo que el de B hacia A, ya que vuelve por el mismo camino-.)

Además, es importante comentar que un camión puede cargar suministros de los nodos por el que este pasa; por ejemplo, un camión que va del nodo A al C y pasa por el B, en caso de ser necesario puede cargar suministros del nodo B al pasar por el mismo (obviamente para realizar esto el nodo B debe tener suministros y el camión debe tener capacidad para cargarlos).

Teniendo en cuenta lo anteriormente mencionado, a continuación se definen los principales elementos que definen la estructura del modelo:

1. Mapa Logístico (conjunto de nodos y aristas con sus características asociadas)
 - $N = \{1, \dots, n\}$: conjunto de nodos (demandantes, proveedores e intermedios)
 - A : conjunto de aristas entre los nodos i y j , siendo $i, j \in N$
 - d_{ij} : distancia del camino entre el nodo i y j siendo $(i, j) \in A$, medido en una unidad de distancia (por ejemplo kilómetros)
 - v_{ij} : velocidad promedio que se puede desarrollar en la arista (i, j) , depende del estado del camino y del tipo de ruta (pedregullo, asfalto, etc.). Se supone que los vehículos que transitan una arista van a la velocidad que indica la misma.
 - p_{ij} : confiabilidad del camino entre los nodos i y j . Se refiere a la probabilidad de que el vehículo complete de forma satisfactoria el camino.

- : probabilidad de saqueo en la arista (i,j)

Estos últimos dos valores son muy difíciles de calcular, y muchas veces son basados en percepciones subjetivas de “expertos”.

2. Productos

- : cantidad de suministros en el nodo $i \in N$ en unidades de suministros.
 - : demanda en el nodo $i \in N$ en unidades de suministros.
 - : cantidad global a ser distribuida en la operación en unidades de suministros.
- Notar que para los nodos intermedios.

3. Vehículos

- K : cantidad de tipos de vehículos diferentes.
- : capacidad del vehículo de tipo k en unidades de suministros.
- : costo de transportar una unidad de suministro desde el nodo i al j , siendo $(i,j) \in A$ (costo por unidad de suministro y distancia). Generalmente, es el costo asociado a la gasolina.
- : costo de reparación de un vehículo del tipo k utilizado en el camino (i,j) (costo por unidad de distancia). Este costo corresponde, generalmente, al pago de salarios del chofer y costo de transportar vehículos vacíos.
- : disponibilidad de vehículos del tipo k en el nodo $i \in N$

4. Presupuesto para llevar a cabo la operación, b .

Se considerarán las siguientes variables de Decisión:

- : número de vehículos del tipo k usados en el link $(i,j) \in A$
- : cantidad de suministros llevados desde i a j
- : cantidad de suministros que se encuentra en el nodo i al final de la operación.

Variables auxiliares:

- : variable binaria que tiene el valor 1 si algún vehículo está viajando del nodo i al j , sino vale 0
- : variable continua para la eliminación de sub-ciclos

El modelo debe satisfacer las siguientes restricciones duras (restricciones originales del problema y de obligatorio cumplimiento):

- Correspondencia de carga en cada nodo:

(1)

- Límite de transporte de suministros a un nodo:

(2)

- Correspondencia de vehículos en cada nodo:

(3)

- Relación vehículos-carga (limita la carga transportada a la capacidad del vehículo):

(4)

- Relación vehículos-carga-uso (fuerza a usar los vehículos de forma racional,

no permitiendo utilizar vehículos que no serán usados). La idea básica aquí es permitir el uso de vehículos vacíos por un arco solamente si luego serán utilizados en otra ruta.

(5)

Vale aclarar que cuando son los costos los que se busca minimizar, nunca se utilizaran vehículos vacíos, sin embargo, cuando lo que se busca minimizar es la probabilidad de saqueo, muy probablemente se utilicen “convoyos” con vehículos vacíos para aumentar la seguridad de la flota; esto va a depender de la prioridad que tengan cada uno de estos objetivos.

- Eliminación de sub-ciclos (se utiliza la formulación de Miller-Trucker-Zemlin [42]):

(6)

- Relación vehículos-uso:

(7)

(8)

- Presupuesto (el modelo asume que los vehículos retornan vacíos por el mismo camino que por el que fueron):

(9)

Las restricciones duras presentadas anteriormente, deben verificarse necesariamente para que el problema sea resoluble; en caso de que dicho conjunto de restricciones no genere ningún punto factible (es decir, de como resultado el conjunto vacío) el problema no tendría solución alguna.

Objetivos del modelo

La meta fundamental que persigue el modelo es entregar la cantidad de suministros existentes “q”, o en su defecto alcanzar en la mayor medida posible el valor “q”. Pero las restricciones duras, por ejemplo las del tipo “cantidad de vehículos disponibles” y “presupuesto existente”, pueden llevar a que no se pueda cumplir en toda su magnitud la meta anteriormente mencionada. En ese caso, lo que el modelo intentará, será enviar la mayor cantidad de suministros posibles.

Se define como una variable de desviación para el objetivo . Dicha variable va a representar las diferencias existentes entre los niveles de aspiración impuestos por el decisor y el resultado que se alcanza en el objetivo.

También se definen objetivos opcionales referentes a factores de ruteo, como ser ajustar del costo al presupuesto, tiempo en recorrer el camino, máxima probabilidad de saqueo en el camino, mínima probabilidad de confiabilidad de la ruta; y referentes a factores de demanda, como ser equidad en la distribución de suministros y prioridad de determinados nodos para recibir suministros.

De esta manera, en el modelo se define la variable , siendo , que representarán los niveles de aspiración de los objetivos anteriormente

mencionados, es decir, lo que se desea alcanzar como mínimo, no superar o alcanzar exactamente.

De esta manera, se definen las siguientes variables asociadas a los objetivos:

- : variable de desviación para el objetivo , estas variables son positivas por definición.
- : variable continua no negativa relacionada a la máxima probabilidad de saqueo en una ruta
- : variable continua no negativa relacionada a la mínima probabilidad de confiabilidad en una ruta
- : máxima desviación de la demanda de los nodos

Finalmente, considerando los objetivos y variables planteadas anteriormente, lo que se busca es determinar un nivel de aspiración para cada objetivo y minimizar las desviaciones de dichos niveles; de esta manera, si se consideran todos los objetivos como un conjunto, lo que se buscará será minimizar la suma ponderada de las desviaciones respecto a los niveles de aspiración

Restricciones blandas (de cumplimiento optativo)

Estas restricciones serán consideradas teniendo en cuenta la ponderación que el decisor resuelva sobre cada una de ellas, por lo tanto, algunas de estas restricciones tendrán mayor incidencia que otras, y se buscará optimizar algunas de estas restricciones sobre otras.

- Como se vio anteriormente, uno de los principales objetivos de la logística humanitaria es suministrar la mayor cantidad de suministros posibles a cada uno de los nodos, para cumplir con este requerimiento se formula la restricción:

(10)

Idealmente, si se logra cubrir toda la demanda, la desviación debe ser igual a cero.

- Muchas veces, el costo es uno de los principales factores a considerar en los sistemas de logística, sin embargo, cuando se hace referencia a la logística humanitaria, generalmente el costo no siempre toma tal importancia (se suelen priorizar otros factores). No obstante, en la logística humanitaria también sucede que generalmente el costo en el que se incurre para poder llevar a cabo la distribución de suministros, es una cota superior sobre la cual la operación debe ajustarse, ya que generalmente la cantidad de dinero con la que se dispone para realizar la logística es sumamente acotada y muchísimas veces insuficiente. De esta manera, se define una restricción, que en base a las opciones tomadas por el decisor, tendrá menor o mayor influencia sobre la solución generada:

(11)

- El tiempo, en términos de logística humanitaria, muchas veces es también un factor crítico a la hora de tomar decisiones. Asimismo, mantener el tiempo por debajo de un valor objetivo es muy común para lograr minimizar los tiempos de traslado de suministros, logrando hacer llegar la

ayuda lo más rápido posible. De esta manera, se define la siguiente restricción para ajustar el tiempo en el que se debe entregar la carga:

(12)

- Muchas veces (principalmente en países subdesarrollados pero no exclusivamente), la operación debe llevarse a cabo en un ambiente hostil o de inseguridad. Por ejemplo, en zonas de guerra, en lugares donde el desastre es tan grande que no existen autoridades que aseguren un mínimo de orden (por ejemplo el caso de Haití), es donde se debe considerar de gran relevancia los factores correspondientes a la seguridad de la operación.

Como se mencionó anteriormente, la probabilidad de saqueo decrece a medida que aumente el número de vehículos que viaje juntos por el mismo arco. Por lo tanto, la probabilidad de saqueo se define como:

A partir de la expresión anterior, utilizando propiedades logarítmicas y multiplicando la expresión resultante por -1 (para trabajar con valores positivos), se busca hallar el valor de probabilidad mínimo de saqueo (PMAX) considerando cada uno de los caminos existentes, para hallarlo se hace:

Se busca el valor mínimo porque, como se mencionó anteriormente, la expresión que indica la probabilidad de saqueo en cada uno de los arcos se encuentra multiplicada por -1 (sino habría que buscar el máximo).

Finalmente, se agregan las siguientes restricciones al modelo:

(*) (13)

(14)

- Siendo "bound" suficientemente grande como para hacer a la expresión * redundante cuando .
- PMAX indica la probabilidad máxima de saqueo que existe en un arco del grafo.
- Otro factor importante a considerar antes de realizar la distribución es la confiabilidad de cada una de las rutas. Los desastres naturales, usualmente, causan gran daño sobre las estructuras (rutas, edificios, etc.), e incluso pueden seguir ocurriendo desastres durante el transcurso de la operación (por ejemplo deslizamientos de tierra producidos por copiosas lluvias), forzando a rediseñar la red de distribución en varias ocasiones.

De esta manera, se define la confiabilidad, la cual indica la probabilidad de cruzar de forma satisfactoria un arco de la red.

También, al igual que en el caso anterior, se busca hallar el valor de confianza máximo (RMIN) considerando cada uno de los caminos existentes a partir de la expresión que define la confianza de un arco.

De esta manera, se agregan las siguientes restricciones al modelo:

(15)

(16)

RMIN indica la confiabilidad mínima existe en los arcos del grafo

- Al realizar el modelo, se debe tener en cuenta el poder lograr una

distribución equitativa de los suministros existentes, ya que si no se considerara este criterio, primero se abastecerían a los nodos que estuvieran más cerca o tuvieran menor costo llegar hasta ellos, y luego se seguiría con el resto de los nodos. De esta manera, en busca de minimizar la desviación entre el suministro a un determinado nodo y la demanda que necesita el mismo, se propone la siguiente restricción para el modelo:

$$\text{---} \quad (17)$$

$$(18)$$

- Algunas veces, y en contraparte al punto anterior, es necesario enfocar y dirigir los suministros principalmente a determinados nodos dentro de la zona afectada, priorizándolos sobre el resto. De esta manera, se plantea la siguiente restricción donde L es el nodo priorizado:

$$\text{---} \quad (19)$$

Siendo L el nodo priorizado.

Notar que los últimos dos objetivos, son mutuamente excluyentes, ya que si se decide distribuir los suministros de manera equitativa ningún nodo tendrá prioridad sobre los otros, lo mismo ocurre en el caso contrario. También existen otros objetivos contradictorios, como ser el caso de minimizar costos con minimizar tiempo (se buscará utilizar medios de transporte más rápidos, que no siempre son los más baratos) y minimizar la probabilidad de saqueo (se tenderá a utilizar vehículos vacíos para aumentar el tamaño del convoy), este último también se contrapone con el objetivo de maximizar la confiabilidad de que lleguen los suministros (se distribuyen los vehículos por diferentes caminos para maximizar la probabilidad de que lleguen al destino).

4.3 IMPLEMENTACIÓN DEL MODELO

Como se mencionó anteriormente, el principal objetivo del modelo desarrollado en [15] es satisfacer de la mejor manera posible la demanda existente en cada nodo, intentando alcanzar el objetivo de entregar todos los suministros existente “q”.

Una vez resuelto el primer nivel, se pasa al segundo, donde se encuentran los 6 restantes objetivos. El cumplimiento de este objetivo se considera de carácter central, de manera que su cumplimiento es priorizado sobre los otros 6 objetivos, por lo que en el modelo se debe definir dos niveles de prioridad, uno que contemple el primer objetivo, y otro que agrupe los 6 restantes. También se pueden definir prioridades (a través de ponderaciones) dentro de estos 6 objetivos; a su vez, se pueden separar estos objetivos en más niveles de prioridad (en el modelo actual solo existen 2 niveles).

De esta manera, a la hora de buscar la solución del problema, se preferirá una

solución que mejore el objetivo del primer nivel y luego los objetivos del segundo nivel (teniendo en cuenta las prioridades existentes en este).

La desviación de estos objetivos, será penalizada y el grado de penalización deberá ser elegido por el decisor en función a la prioridad que éste haya elegido para cada meta. Dicha penalización será incluida en el modelo mediante los parámetros α_i .

Por lo tanto, se tenderá a minimizar el valor de las variables de desviación, este fenómeno se acentúa aun mas en las metas priorizadas.

Finalmente, la función objetivo del problema de programación por metas a minimizar sería:

Sujeto a las siguientes restricciones:

Observando lo mencionado anteriormente, se desprende claramente que el modelo a resolver en el primer nivel es el siguiente:

Si $x_j^* < b_j$ entonces no podrá alcanzarse el objetivo de entregar la cantidad de suministros deseados.

Luego de obtener la solución óptima del primer nivel (x_j^*), el modelo para el segundo nivel es:

La idea de que se cumpla esta última restricción es que se mantenga la meta obtenida en el primer nivel y en base a eso se calculen las metas del segundo nivel.

5 EVACUACIONES

5.1 INTRODUCCIÓN

Habitantes de áreas urbanas pueden encontrarse en peligro frente a la ocurrencia de desastres naturales o causados por el hombre, como ser inundaciones, huracanes, terremotos, incendios, accidentes químicos o nucleares. Para proteger a las personas que allí residen, es necesaria la realización de un plan de evacuación de manera que ante la ocurrencia del desastre se cuente con procedimientos bien definidos de modo que todas las personas sepan cómo actuar para ponerse a salvo. Estos planes pueden abarcar tanto la evacuación a nivel urbano (evacuación de la zona afectada) como la evacuación a nivel de estructuras (evacuación de edificios).

5.2 EVACUACIONES DE ESTRUCTURAS

Existen numerosos estudios y modelos para llevar a cabo evacuaciones a nivel de estructuras, abarcando evacuaciones de edificios, estadios, hospitales, plantas químicas, etc. A su vez, también existen muchos programas informáticos que simulan la ocurrencia de incendios en estructuras y modelan el comportamiento de los planes de evacuación. Además, también existen sistemas que guían a las personas dentro de un edificio para que las mismas tomen rutas de evacuación seguras; este es el caso del sistema MILS, que a través de luces (leds) en el suelo, indica las rutas seguras (con colores verde y rojo) para evacuar el edificio en caso de incendios, evitando el estrés y la confusión que genera en las personas una alarma de incendio, resultando en menores pérdidas humanas [32].

En este tipo de evacuaciones, es de suma importancia que el plan de evacuación sea divulgado a todas las personas de la organización, con el fin de saber qué hacer ante la ocurrencia de una emergencia.

Uno de los riesgos fundamentales en las organizaciones (sea esta de cualquier tipo) son los incendios, ya que es una amenaza existente en todo lugar donde hayan personas desarrollando actividades y es uno de los incidentes más comunes y letales [23].

Para la elaboración del plan, es recomendable formar un equipo multidisciplinario, idealmente compuesto por directivos de la organización, defensa civil, policías, bomberos y médicos. Se debe tomar en cuenta la cantidad de personas que concurren en las distintas horas del día, y conocer las falencias y vulnerabilidades, así como también las amenazas internas y externas a la organización.

Algunos aspectos importantes a considerar son por ejemplo [25]:

- peligro de incendio, derrumbe o inundación

- corredores, pasillos, escaleras y salidas adecuadas para circular rápidamente
- cantidad y ubicación de extintores o mangueras, botiquines, escaleras, herramientas, etc.
- sistemas de alarma

Finalmente, es importante distribuir a lo largo del edificio planos que indiquen las rutas de evacuación hacia las zonas seguras y capacitar al personal para que los mismos sepan cómo actuar ante una emergencia. Además, es altamente recomendable, realizar simulacros para practicar la evacuación, evaluar el plan y corregirlo en caso de ser necesario [25].

5.3 EVACUACIÓN DEL ÁREA URBANA

Para proteger a los habitantes de una determinada área urbana, se deben establecer rutas de evacuación hacia refugios o zonas seguras, de manera que la zona afectada pueda ser despoblada lo más rápido y de la forma más segura posible.

Con tal fin, en primera instancia se deben investigar vulnerabilidades y riesgos existentes en la zona, analizar datos históricos y determinar la ocurrencia con la que suceden incidentes. También es importante implementar dispositivos de vigilancia y sistemas de alertas, de manera que la población pueda actuar antes de que ocurra el desastre (si es posible) reconociendo la alerta correspondiente [24].

Desafortunadamente, no siempre es sencillo contar con personal capacitado o disponer de información calificada para evaluar vulnerabilidades y amenazas de una determinada zona.

Lo que sucede en algunas ocasiones es que la comunidad científica y de ingeniería realizan actividades de investigación y monitoreo teniendo en cuenta principalmente intereses personales; un caso típico de lo anterior es cuando se elige estudiar un determinado volcán por su riqueza científica y no por su proximidad a una zona poblada. Además, la información recabada muchas veces es publicada en revistas científicas empleando un lenguaje y términos no comprensibles por los encargados del manejo de amenazas [24].

En la siguiente sección (punto 5.4), se desarrollará en detalle uno de los planes existentes para la evacuación de zonas urbanas. Vale aclarar que existen diversos tipos de planes para zonas urbanas, el que se analizará trata sobre la evacuación a pie de los habitantes hacia los refugios previamente definidos.

Por otra parte, también existen planes donde se toma por supuesto que cada habitante (o familia) cuenta con un automóvil con el cual pueda realizar la evacuación.

En relación a esto, Brentschneider et al. [29] plantean un modelo de evacuación

que consiste en generar una red de caminos para la evacuación desde la zona afectada hacia los refugios (definidos previamente), de esta manera, el modelo planteado modifica los sentidos de las calles (por ejemplo calles de doble sentido a un solo sentido) para lograr reducir el tiempo de evacuación al mínimo, distribuir la carga de tráfico y así evitar embotellamientos.

Asimismo existen modelos que se utilizan para testear la eficiencia de planes de evacuación, un ejemplo de ello es el modelo desarrollado por Siomonovic et al. [30] en el cual se toman en cuenta factores del comportamiento humano (evalúa planes de evacuación ante inundaciones).

También, es importante mencionar que existen algunos programas informáticos que sirven a los fines de planificar evacuaciones urbanas, uno de ellos es el simulador SimWalk PRO; el mismo analiza los flujos peatonales, detecta congestiones críticas, cuellos de botella y densidades excesivas determinando tiempos de evacuación según los datos ingresados, a la vez de proponer soluciones para minimizar dichos tiempos [34].

5.3.1 PLAN DE EVACUACIÓN - ZONAS URBANAS

El modelo que se expondrá a continuación fue desarrollado en el departamento de Ingeniería Civil de la Universidad de Coimbra (Portugal) [28]. En el mismo, se presenta un modelo multi-objetivo enfocado en determinar rutas para la evacuación urbana (desde las zonas afectadas hacia los refugios), obteniéndose una ruta principal y otra secundaria (en caso de que la primera no pueda ser transitada). Vale aclarar que la ubicación de los refugios debe ser determinada previamente (existen estudios específicos para determinar las ubicaciones más convenientes), utilizándose en el modelo la ubicación de los mismos como entrada.

Los objetivos que se pretenden optimizar son minimizar la distancia del camino principal y secundario que deben recorrer las personas para llegar hasta los refugios, buscar los caminos que tengan menor riesgo para las personas que los transiten, minimizar el riesgo asociado a las zonas donde se encuentran ubicados los refugios, minimizar en lo posible la cantidad de refugios a utilizar, y seleccionar los refugios de manera que el tiempo necesario para transportar a las personas al hospital más cercano sea mínimo.

Estos objetivos, van asociados unos con otros y muchas veces pueden contraponerse (por ejemplo: camino más corto vs camino más seguro), de manera que la interacción entre dichos objetivos son los que arrojarán las mejores rutas (una principal y otra secundaria) que unirán las diferentes zonas afectadas con los refugios seleccionados. También, es importante mencionar que se le debe asignar mayor importancia al camino principal (buscando minimizar tanto la distancia como el riesgo del camino), y que una vez que el camino principal es intransitable, se considera el camino secundario; vale aclarar que ambos caminos deben de ser disjuntos, no compartiendo arcos ni nodos entre sí.

Para lograr una simplificación de la realidad, las zonas a evacuar se dividen en sectores, y los caminos desde cada uno de los sectores hacia los refugios se asumen desde el centro del sector.

El modelo también considera una cantidad máxima de personas que pueden asociarse a un refugio, así como también una cantidad mínima necesaria para abrir el refugio.

Cada sector a evacuar es asociado a un plan de evacuación, donde cada plan se define como i siendo:

- el identificador del plan de evacuación ($i = 1 \dots K$ donde K es el numero total de planes a ser considerados en el análisis)
- el identificador del sector a evacuar asociado al plan i (cada sector debe estar asociado al menos a un plan de evacuación)
- la ruta de evacuación primaria asociada al plan
- la ruta de evacuación secundaria asociada al plan

Finalmente, un plan de evacuación será asociado a cada sector a evacuar.

Teniendo en cuenta lo anteriormente mencionado, a continuación se definen las variables que se utiliza en el modelo presentado:

- variable binaria, siendo 1 si el plan i es elegido y 0 en caso contrario
- variable binaria, siendo 1 si el refugio j es elegido y 0 en caso contrario
- número de sectores a evacuar (parámetro entero)
- número de refugios existentes (parámetro entero)
- refugio asociado al camino principal (p_i) del plan i
- refugio asociado al camino secundario (s_i) del plan i
- distancia del camino p_i (parámetro real)
- distancia del camino s_i (parámetro real)
- riesgo asociado al camino p_i
- numero de personas a ser evacuadas del sector k (parámetro entero)
- cantidad de personas que puede soportar el refugio j (parámetro entero)
- cantidad mínima de personas necesarias para abrir el refugio j (parámetro entero)
- tiempo requerido para evacuar a las personas del refugio (hacia un hospital) asociado al plan i (parámetro real)
- riesgo asociado al refugio del plan i (parámetro real)

Seguidamente, se detallarán las funciones objetivos a optimizar:

- minimizo el largo de las rutas principales
- minimizo el riesgo asociado a las rutas principales
- minimizo el largo de las rutas secundarias
- minimizo el riesgo asociado a los refugios
- minimizo el tiempo necesario para realizar la evacuación desde los refugios a los hospitales
- minimizo la cantidad de refugios necesarios

Además se deben satisfacer las siguientes restricciones:

- con
Se elige un plan de evacuación para cada sector, siendo la cantidad de sectores a evacuar, por lo que se tendrán restricciones.
- con
Se controla que la capacidad máxima del refugio no sea excedida, siendo la cantidad de refugios existentes, por lo que se tendrán restricciones.
- con
Se asegura que la cantidad de personas asignadas al refugio sea al menos la mínima necesaria como para poder abrirlo, siendo la cantidad de refugios existentes, por lo que se tendrán restricciones.
-
-

Vale aclarar que el modelo toma como dadas un conjunto de rutas desde cada sector hacia los refugios existentes (pudiendo existir varias rutas entre un sector y un refugio dado); lo que se hace aquí es, a partir de dicha información (rutas existentes), ubicación de los refugios potenciales y obteniendo datos propios del desastre (como posibilidad de transitar un determinado camino), se determinan los caminos de evacuación más convenientes para cada una de las zonas.

Igualmente, existen algoritmos para calcular el camino más corto entre dos nodos pertenecientes a una red (por ejemplo el algoritmo de Dijkstra). También existen numerosos algoritmos para encontrar caminos disjuntos entre dos pares de vértices de una red [26].

6 ALBERGUES TEMPORALES

6.1 INTRODUCCIÓN

El objetivo principal de los albergues temporales es proteger la vida de las personas, reduciendo así el número de víctimas de la sociedad afectada así como también el sufrimiento de las mismas. A su vez, se deben asegurar condiciones dignas para los sujetos que allí residan, atendiendo las necesidades básicas de cada una de las personas y brindando los servicios básicos necesarios, asegurando la estabilidad tanto física como psicológica de los refugiados [40].

Se debe contar con los recursos necesarios para proporcionar alimento, vestimenta y salud a las personas, durante el periodo que sea necesario luego de ocurrido el fenómeno destructivo. Otros aspectos importantes a considerar son el saneamiento (involucrando desde el manejo y suministro de agua potable hasta el manejo y la eliminación de excretos y residuos), la coordinación de los suministros y el cuidado de alimentos, bienes y recursos [40].

Algunas recomendaciones establecidas por la Organización Mundial de la Salud a cumplirse en los albergues son por ejemplo ofrecer 2500 calorías por día por persona (mínimo) y 18 litros de agua diarios (mínimo) para beber y aseo personal, así como disponer de habitaciones de 4 metros cuadrados por persona. [40]

Los albergues deben contar con una infraestructura segura, siendo capaces de proteger contra condiciones ambientales adversas, a la vez que deberán estar ubicados en terreno seguro [39].

La localización de los mismos se debe establecer en base al conocimiento que se tenga sobre las zonas propensas a ser afectadas por los potenciales desastres a considerarse. En caso de que la construcción o elección de los albergues se realice una vez ocurrido el desastre, se deben considerar las zonas que no estén afectadas y que difícilmente lo estén mientras dure el desastre (por ejemplo terrenos altos en caso de inundaciones) [40].

A modo general, algunos de los atributos que se buscan en los terrenos a ubicar albergues son por ejemplo un buen drenaje del lugar (se recomienda que la inclinación del terreno no debe sobrepasar los 6% ni estar por debajo del 1% para garantizar el drenaje), que posea garantías mínimas frente a condiciones adversas del clima o meteorológicas (por ejemplo zonas propensas a inundaciones, deslizamientos o avalanchas), encontrarse cerca de una carretera para facilitar el abastecimiento de suministros (generalmente se opta por localizaciones fuera del área urbana para facilitar el acceso) y en caso de esperar recibir ayuda internacional buscar localizaciones cerca de un aeropuerto o puerto. También se debe buscar, en la medida de lo posible, el cuidado y la conservación ambiental, no

afectando el medio ambiente al ubicar los albergues (por ejemplo tala de árboles); lo ideal sería que los gobiernos, al desarrollar sus planes de reordenamiento territorial especifiquen posibles zonas para la implantación de albergues temporales [38].

Existen distintos tipos de albergues temporales que se utilizan en la práctica; la utilización de unos u otros generalmente esta impuesto por la complejidad o severidad de la situación de emergencia, así como la capacidad y recursos locales existentes, los tipos de albergues más comúnmente utilizados definidos en [38] son:

- Auto-albergue: cuando las personas afectadas deciden ubicarse en casas de familiares o amigos
- Comunitarios – infraestructura instalada: son los más comunes en una primera instancia (principalmente en lugares donde no existen refugios previamente construidos). Se utilizan lugares públicos (escuelas, gimnasios, centros comunales, etc.) para ubicar a las personas evacuadas y su ubicación se difunde a través de los medios de comunicación. Los mismos deben ser acondicionados y su utilización es transitoria. El problema que existe con este tipo de albergues es que si se prolonga su utilización en el tiempo se limita la posibilidad de volver a utilizar el edificio con el fin para el cual fue concebido, retrasando la recuperación de la sociedad; un claro ejemplo de esto son las escuelas o edificios estatales.
- Comunitarios – campamentos de emergencia: de utilización temporal a corto plazo se utilizan tiendas de campaña, carpas o materiales livianos para alojar a los evacuados. Requieren de gran administración además de demandar especial control a nivel sanitario y de seguridad.
- Fijos: son construcciones específicas para utilizarse como albergues que cuentan con servicios necesarios para la permanencia de las personas por un periodo prolongado. Este tipo de albergues se encuentran en zonas donde la ocurrencia de desastres es habitual, como ser lugares ubicados en las laderas de montañas donde el desplazamiento de tierra provocado por las lluvias es usual, o ubicados cerca de volcanes activos.
- Subsidios de arriendo: si bien no es la alternativa más común, existen ocasiones en las que el estado les brinda a los afectados (o algunos de ellos) recursos económicos para que puedan costear el alquiler o arriendo de una vivienda que garantice las condiciones mínimas de vivienda.

Generalmente, ante la ocurrencia de un desastre de grandes magnitudes, es necesaria la utilización de una combinación de estos tipos de albergues en un mismo sector [38].

Es responsabilidad de las autoridades el adecuado manejo administrativo, técnico y operativo de los albergues, debiéndose emplear personal específico para el manejo de cada uno, ya que generalmente no todos funcionan de la misma manera. Por ejemplo, la planificación, apertura, mantenimiento y cierre de cada albergue temporal es diferente, debido a factores como el número de refugiados, infraestructura disponible, aspectos socio-culturales, etc.; estos temas deben ser manejados por el personal responsable de cada uno de los refugios. El mismo debe estar compuesto básicamente por trabajadores sociales, psicólogos y doctores (se

debe tener un registro para contactar a los mismos y poder convocarlos rápidamente) [38].

Una vez abierto un albergue, su tiempo de funcionamiento va a depender en gran medida de la gravedad e intensidad del fenómeno particular y de la capacidad de recuperación local. Una vez que no existe más peligros asociados al desastre y existan las condiciones mínimas de habitabilidad y seguridad en las viviendas de los damnificados, los mismos podrán abandonar el albergue. Sin embargo, es muy común que los albergues temporales tiendan a prolongar su operación por más tiempo que el necesario, generándose una situación de dependencia por parte de los allí residentes difícil de superar, por lo tanto, es imperativo que las autoridades fomenten la desocupación del albergue (cuando las condiciones lo ameriten) tratando de volver a la normalidad lo más rápidamente posible [40].

Como se mencionó anteriormente, la localización de los albergues debe realizarse de forma personalizada en cada región, existiendo normas y recomendaciones generales a seguir pero finalmente debiendo analizar de forma individual cada caso en particular. En este tipo de decisiones juegan un papel fundamental los Sistemas de Información Geográfica, ya que a partir de ellos se puede analizar en detalle la zona, identificar regiones seguras y realizar simulaciones de los fenómenos más comunes que afectan el área, pudiéndose sacar conclusiones muy provechosas. También es importante tener conocimiento de los fenómenos que en el pasado han golpeado la zona así como también la gravedad y los resultados que los mismos produjeron [40].

Actualmente, existen algunos sistemas útiles para las personas afectadas a la hora de buscar la ubicación de los albergues temporales. Uno de estos sistemas es el Sahana [21], el cual es un software web que, entre otras funcionalidades, provee la ubicación de los albergues temporales en un mapa y también brinda información acerca de los mismos (por ejemplo capacidad y número de personas albergadas). Además permite que las personas ingresen su ubicación (albergue donde residen) y a través de un buscador se puede consultar la ubicación de otra persona (útil si se desea saber sobre un ser querido).

Por otra parte, es posible utilizar la plataforma de Google Maps[] para ingresar la ubicación de los albergues temporales, de manera que cualquier persona con un dispositivo móvil e internet tenga acceso a dicha información. Además, también existen algunas aplicaciones para móviles que pueden ser de ayuda en este tipo de situaciones [91], algunas de ellas son Buddy Guard [89] la cual envía nuestra ubicación (coordenadas) a una lista de contactos de emergencia, y otra es Pocket First Aid & CPR [90] la cual le brinda al usuario información básica sobre cómo actuar y que hacer en ocasiones de emergencia (como por ejemplo distintos tipos de desastres naturales).

A continuación se presentará un pequeño modelo matemático que, a partir de las posibles zonas seguras donde instalar los albergues, se deciden cuáles son las más convenientes y cuantos albergues corresponde abrir (respetando las capacidades mínimas y máximas de personas por albergue).

6.2 MODELOS MATEMÁTICOS

Uno de los aspectos más importantes a la hora de manejar una situación de emergencia es la de seleccionar que refugios deben abrirse (o levantarse) para atender toda la demanda existente por parte de los afectados, de manera de no mal utilizar o desperdiciar recursos (tanto humanos como materiales), los cuales generalmente no abundan en estas situaciones.

El modelo presentado a continuación fue desarrollado por el departamento de Ingeniería Civil de la Universidad de WanzhouIngenieria [35]; en el mismo, se establece una red donde los nodos son las zonas a evacuar y los posibles lugares donde establecer los refugios. Además, la distancia entre cada uno de los nodos de la red debe estar previamente definido, estableciéndose un límite de distancia a la hora de realizar la asociación entre las zonas afectadas y los refugios. También, es necesario conocer la cantidad de personas a evacuar de cada una de las zonas afectadas.

A continuación se presentan las variables a utilizar en el modelo:

- distancia máxima aceptable entre una zona a evacuar y el albergue correspondiente (umbral)
- 1 si la distancia del nodo i (zona afectada) al nodo j (albergue) es menor que d_{ij} , y 0 en caso contrario
- cantidad de personas a ser evacuadas del nodo i
- cantidad de albergues que se desean abrir
- 1 si el nodo i esta asociado a uno o mas refugios, y 0 en caso contrario
- 1 si el albergue del nodo j es utilizado, y 0 en caso contrario
- capacidad máxima de personas que puede aceptar el albergue situado en el nodo j
- capacidad mínima de personas que es necesaria para abrir el albergue situado en el nodo j
- cantidad de albergues disponibles
- cantidad de zonas a evacuar

La formulación matemática a este problema se plantea a continuación:

- Se maximiza la cantidad de personas a establecerse en los albergues. Idealmente, la cantidad de albergues es la suficiente y la distancia de los mismos a las zonas afectadas está por debajo del umbral establecido.
- Sujeto a las restricciones:
 - Se establece que los evacuados del nodo i deben alojarse entre los albergues cuya distancia es menor a d_{ij} .

- Establece que la cantidad de albergues a abrirse sea menor o igual que la cantidad de albergues especificados (número de albergues que se desean abrir). Si se especifican la cantidad máxima de albergues disponibles () se seleccionarán aquellos que sean necesarios para cubrir en la mayor medida posible el número de evacuados.
- Se consideran los refugios que tengan una distancia menor o igual del umbral establecido () a las zonas afectadas.
- Se limita el número de personas en el albergue a la capacidad máxima del mismo
- Se establece que el albergue debe llegar a la capacidad mínima necesaria para abrirse
-
-

A continuación se presenta otro modelo, desarrollado por Mejía et al. [82], el cual resulta ser un poco más complejo que el último mencionado. El mismo contempla varios aspectos de la realidad como suministros, presupuestos, modos de transporte, niveles de ocupación y costos entre otros, abordando además cuestiones que tienen que ver con la gestión de inventarios y distribución de suministros.

Tiene como objetivos encontrar la cantidad de bienes a ser suministrados, las rutas de abastecimiento a los albergues desde los centros de distribución e identificación de las zonas a ser evacuadas, estando principalmente orientado a desastres naturales del tipo inundaciones.

A continuación se presentan las variables a utilizar en el modelo:

- Porcentaje mínimo de ocupación de un albergue requerido para autorizar su apertura
- Nivel mínimo de utilización conjunta de la capacidad de los centros de distribución
- Presupuesto designado para las actividades de prevención frente a una inundación
- Presupuesto asignado para la reacción frente a una inundación
- Porcentaje de la carga transportada por las ONG's
- Número máximo de centros de distribución a abrir definido por las autoridades
- Costo fijo de la habilitación del centro de distribución i
- Costo unitario de adquisición del producto l
- Costo de apertura del albergue j

- Capacidad en personas del albergue j
- Número de personas a albergar que se encuentran en cada zona de demanda k
- Factor de conversión de cada producto l acorde a la unidad de distribución
- Peso unitario en kilogramos del producto l
- Volumen unitario en metros cúbicos del producto l
- Capacidad de cada centro de distribución i, en metros cúbicos
- Capacidad de cada modo de transporte m, en toneladas.
- Peligro asociado a cada zona de demanda k, determinado en la parte geográfica ($0 \leq \dots \leq 1$)
- Distancia existente en km entre cada zona de demanda k y cada albergue j
- Distancia existente en km entre cada centro de distribución i y cada albergue j
- Costo unitario de transporte por vehículo del cada centro de distribución i a cada albergue j, por cada modo de transporte m
- Costo unitario de transporte por tonelada de cada centro de distribución i a cada albergue j, por el modo de transporte m
- Matriz de conectividad de cada modo de transporte m para ir del centro de distribución i a cada albergue j. Se asigna 1 si para el par de instalaciones (i, j) existe conectividad terrestre para el modo m y 0 en otro caso

Variables de decisión:

- Cantidad de productos tipo l enviados de cada centro de distribución i a cada albergue j
- Cantidad de inventario del producto l pre posicionado en cada centro de distribución i
- Número de toneladas de producto transportadas de cada centro de distribución i a cada albergue j
- Número de toneladas de producto transportadas por modo de transporte m de cada centro de distribución i a cada albergue j
- Número de personas que se encuentran asignadas a cada albergue j
- Número de viajes por modo m enviados del centro de distribución i a cada albergue j
- Número de toneladas de ayuda humanitaria enviada por modo de transporte m, de cada centro de distribución i a cada albergue j
- Número de personas residentes en cada zona de demanda k que se dirigen a cada albergue j
- Si el centro de distribución i es habilitado, se hace uno. En otro caso, es cero
- Si el albergue j es habilitado, se hace uno. En otro caso, es cero.

Funciones objetivo:

-

-

Restricciones:

- Gestión de evacuación:
 -
 -
- Respetar capacidades en centros de distribución y albergues:
 -
 -
 -
- Determina el número máximo de centros de distribución a ser abiertos:
 -
- Controlar la gestión de ayuda humanitaria por ONG`s y por parte del gobierno:
 -
 -
 -
 -
 -
 -
- Controlar presupuestos:
 -
 -

En [82] también se presenta otro modelo muy interesante que se enfatiza en la identificación de la ruta de evacuación que se debe seguir por cada modo, en cada

par (zona a evacuar, albergue).

7 RELEVAMIENTO DE HEURÍSTICAS Y METAHEURÍSTICAS PARA LA DISTRIBUCIÓN DE SUMINISTROS

En la literatura, existen números modelos que plantean el problema de distribución de suministros desde depósitos a clientes finales. A su vez, existe una tendencia de agregarles a los modelos cada vez más características de la realidad, generando modelos más aplicables, pero a su vez también dificultando significativamente la resolución de los mismos aumentando la complejidad del problema.

Estos problemas son de optimización combinatoria (perteneciendo en su mayoría a la clase de problemas NP-Hard), creciendo exponencialmente el número de combinaciones con el tamaño del problema; por lo tanto no se pueden resolver en tiempo polinomial en función del tamaño de la entrada, pues la cardinalidad del espacio de búsqueda suele ser demasiado grande como para resolver el problema con algoritmos exactos (demasiado tiempo necesario).

De esta forma, se hace necesaria la búsqueda de algoritmos que permitan resolver dichos problemas de manera eficiente, obteniendo una solución de calidad en un tiempo razonable.

Para tal fin, se utilizan heurísticas y metaheurísticas que resuelvan los problemas de ruteo de vehículos.

Dos de los problemas más conocidos de este tipo son el TSP (Traveler Selsman Problem) y el VRP (Vheicle Routing Problem).

7.1 TSP

El problema TSP consiste en un conjunto de ciudades que deben ser visitadas por un viajero. El mismo parte desde un punto origen, debe pasar exactamente una sola vez por cada ciudad y finalmente regresar al punto de inicio, de manera de minimizar la distancia total (o bien tiempo o costo).

Una de las heurísticas más básicas para el problema TSP es la Heurística del vecino más cercano. La misma consiste en ir eligiendo en cada paso como el nodo siguiente al nodo más cercano al nodo actual, partiendo desde el origen (deposito). De esta manera, se van visitando todos los clientes y se forma la ruta solución [43].

7.2 VRP

Los problemas VRP son un conjuntos de problemas donde lo que se intenta es satisfacer la demanda de los nodos hallando un conjunto de rutas óptimas para realizar el reparto de manera de minimizar los costos [45].

Existen muchas variantes del VRP, la más clásica y conocida es el CVRP (Capacitated VRP). La misma consiste en un conjunto de nodos con determinadas demandas y un nodo de origen con uno o varios vehículos los cuales tienen una capacidad asociada (la capacidad es homogénea para todos los vehículos). El objetivo es encontrar el conjunto de rutas solución (una para cada vehículo) para realizar la distribución (los vehículos deben salir del nodo origen y regresar al mismo) de manera de respetar las capacidades de los vehículos, visitar cada nodo una sola vez y satisfacer en la mayor medida posible la demanda de los nodos (siempre intentando minimizar costos) [57].

Una variante del CVRP es el FSMVRP (Fleet Size and Mix VRP), donde se cuenta con una flota de vehículos heterogénea, de manera que las capacidades de los vehículos, sus costos fijos y demás atributos (por ejemplo velocidad si se considerase) son características individuales de cada tipo de vehículo [57].

Otras variantes del VRP que son de interés analizar en el presente proyecto son el CVRPTW (Capacitated VRP with Time Windows) y el MDVRP (Multi Depot VRP). El primero (CVRPTW) consiste en una variante del CVRP, donde además de las capacidades, cada cliente tiene una ventana de tiempo asociada y debe ser atendido por un vehículo en el intervalo de tiempo indicado por la ventana. Por lo tanto, los vehículos deben visitar a los clientes dentro del intervalo de tiempo indicados por su ventana, y si llegan antes deben esperar al inicio de la venta de tiempo del cliente (ningún vehículo puede llegar al cliente luego de que su ventana de tiempo finalizó) [57].

Por su parte, en el MDVRP existen un conjunto de depósitos y clientes dispersos geográficamente, y cada depósito tiene asociado un conjunto de vehículos (con capacidades homogéneas). El objetivo es determinar una serie de rutas para los vehículos de cada depósito de manera de satisfacer la demanda de cada cliente y minimizar los costos (generalmente se impone la restricción de que cada cliente puede ser visitado por un solo vehículo) [54].

7.2.1 HEURÍSTICAS

A continuación, se definen algunas de las heurísticas más conocidas para resolver los distintos tipos de problemas de VRP planteados anteriormente.

7.2.1.1 ALGORITMO DE AHORROS DE CLARKE Y WRIGHT

Una de las heurísticas más conocidas para el problema de CVRP es el Algoritmo de Ahorros de Clarke y Wright [44] [45].

Se parte de una solución inicial y se busca combinar dos rutas diferentes en una sola (respetando las restricciones del problema, por ejemplo capacidad del vehículo), de manera de disminuir la distancia recorrida. Las soluciones obtenidas con este algoritmo generalmente pueden ser mejoradas con operadores de búsqueda local.

Existen algunas extensiones del algoritmo original propuesto por Clarke y Wright que resuelven algunas de las variantes del VRP.

Solomon [46] propone una extensión del Algoritmo de Ahorros para solucionar problemas con ventanas de tiempo (CVRPTW).

A su vez, Golden et al. [47] incorporan al algoritmo original la capacidad de trabajar con una flota de vehículos heterogénea (FSMVRP), asociándole a cada tipo de vehículos atributos como costo fijo y capacidad.

7.2.1.2 HEURÍSTICAS DE INSERCIÓN

Las heurísticas de Inserción consisten en agregar sucesivamente nodos a las rutas existentes de manera de ir creando la solución (siempre respetando las restricciones del problema). La selección del nodo a agregar puede ser aleatoria o seguir un determinado patrón (por ejemplo nodo más cercano al último nodo agregado o ruta modificada) siempre eligiendo entre los nodos que aún no han sido agregados a ninguna ruta [48].

Pueden realizarse inserciones secuenciales o en paralelo, en las secuenciales el nuevo nodo a insertar se agrega a la última ruta creada utilizando algún criterio de inserción (por ejemplo mejor posición para insertarlo dentro de la ruta –minimizar distancia-) [49]; mientras tanto, en las inserciones paralelas, un cliente se puede insertar en cualquiera de las rutas de la solución.

Solomon plantea una heurística de inserción secuencial para resolver el problema con ventanas de tiempo (VRPTW) [46]; y a su vez, Potvin y Rousseau plantean una versión de inserción en paralelo [50] basado en la heurística desarrollada por Solomon.

La principal desventaja que tienen las inserciones en paralelo es que los últimos nodos a insertar (nodos no visitados aún) tienden a estar dispersos, resultando las últimas rutas construidas con un costo muy elevado; con las heurísticas en paralelo este problema se resuelve [51].

7.2.1.3 CLUSTER FIRST - ROUTE SECOND

Los métodos cluster first - route second (asignar primero – rutear después), como su nombre lo indica, constan de dos fases; en la primera se busca crear clusters

(grupos de clientes) que formaran parte de una misma ruta (respetando las restricciones del problema, por ejemplo capacidad del vehículo), y luego se crea la ruta para cada uno de los clusters creados [51].

Una de las heurísticas más conocidas de este tipo es la Heurística de Barrido (o Sweep) [52], la cual consiste en crear los clusters con los clientes realizando una semirrecta con origen en el depósito e ir girando dicha semirrecta sobre su eje de manera de ir incorporando los clientes “barridos” por dicha semirrecta (respetando las restricciones de capacidad).

Solomon también desarrolla un heurística del Algoritmo de Barrido considerando ventanas de tiempo (CVRPTW) [46]; al momento de formar los clusters (barrido) solamente se consideran restricciones de capacidad, y luego al crear la ruta dentro de cada cluster sí se consideran las restricciones temporales de cada nodo.

Otra heurística importante es la que proponen Fisher y Jaikumar [53].

La misma define en primera instancia un conjunto con algunos clientes donde cada cliente formará parte de un cluster distinto. Luego, resolviendo un Problema de Asignación Generalizada (GAP) se asocian los clientes restantes a los clusters existentes.

Franco e Islas [54] proponen una implementación del cluster first – route second para resolver el problema de MDVRP. Para cada cliente se calcula el costo de asignarlo a cada uno de los depósitos, luego se utiliza una función para asociar los clientes a los depósitos que considera el valor calculado para cada cliente, la demanda que puede satisfacer el depósito, y la cantidad de vehículos del depósito; de esta manera quedan asignados los clientes a los depósitos. Finalmente se procede a realizar el enrutamiento en cada depósito (para los clientes asociados al mismo); el enrutamiento se realiza de manera aleatoria (para una mejor exploración del espacio de búsqueda y evitar caer en óptimos locales), luego se utilizan algoritmos para mejorar el enrutamiento de cada depósito.

Otro trabajo que se basa en la estrategia de cluster first - route second para atacar el problema de MDVRP es el que desarrollan Surekha y Sumathi [55]; en el mismo se utiliza un Algoritmo Genético para hallar la solución [punto 7.2.2.4 - Algoritmos Genéticos]. La asignación de los clientes a los depósitos se realiza en función de la distancia entre ellos (distancia Euclidiana). Una vez asignados los clientes a los depósitos se realiza el ruteo de los vehículos utilizando el Algoritmo de Ahorros de Clarke y Wright en cada depósito, buscando minimizar las distancias necesarias y cantidad de vehículos a utilizarse. Finalmente, en las etapas de cruzamiento y mutación del algoritmo genético, se toma como entrada la solución hallada en el paso anterior y se realiza una exploración del espacio de soluciones buscando obtener una mejor solución.

7.2.1.4 ROUTE FIRST - CLUSTER SECOND

El método rutear primero – asignar después fue propuesto inicialmente por Beasley [56], y al igual que en el caso anterior, también consta de dos fases. En la

primer instancia se crea una ruta que visite todos los nodos (sin considerar las restricciones del problema, como por ejemplo la capacidad del vehículo) de manera similar a como se resuelve un TSP. Luego, se busca la mejor partición de dicha ruta de manera de respetar las restricciones del problema y se obtienen las rutas solución.

7.2.1.5 BÚSQUEDA LOCAL

Los procedimientos de Búsqueda Local consiste en mejorar una solución existente buscando soluciones vecinas que mejoren el valor de la función objetivo; este procedimiento se repite hasta no encontrar ninguna solución vecina que mejore el valor de la solución actual; por lo tanto se obtiene una solución localmente optima respecto a su vecindad [57].

En el contexto del ruteo de vehículos, las búsquedas locales pueden aplicar dentro de una de las rutas solución (por ejemplo modificar el orden en que se visitan los nodos en una ruta), o entre varias rutas (pudiéndose intercambiar conjuntos de nodos entre distintas rutas, además del orden de visita de los nodos dentro de una ruta) [57].

Uno de los procedimientos de búsqueda local más conocidos para la optimización de una ruta es el λ -intercambio [58], el cual consiste en eliminar λ (valor fijado de antemano) arcos de la ruta y reconectar los nodos afectados (nodos extremos de los arcos eliminados) entre sí de manera de generar λ nuevos arcos en la ruta y mejorar la solución; esto puede tener como consecuencia que se modifique el orden en que se visitan los nodos.

Se denomina λ -opt al procedimiento de búsqueda local que utiliza λ -intercambios, y se dice que una solución es λ -óptima cuando no se puede mejorar más realizando λ -intercambios [59], generando un óptimo local.

La heurística de Lin-Kernighan [60] es similar al procedimiento de búsqueda local λ -intercambio en la medida que también se basa en el intercambio de conjuntos de arcos en la ruta (intercambios λ -opt que generan soluciones λ -óptimas); la principal diferencia radica en que la cantidad de arcos a intercambiar en esta heurística (λ) es variable, determinándose dinámicamente.

Bautista y Pereira [59] plantean una implementación de la heurística de Lin-Kernighan adaptada para el problema de CVRP.

El procedimiento Or-opt consiste en eliminar un conjunto de “k” nodos consecutivos de la ruta y ubicarlos en otra posición dentro de la misma ruta de manera de formar un circuito cerrado de coste menor al original. Primero comienza reordenando conjuntos de 3 nodos ($k=3$) consecutivos hasta obtener la solución 3-óptima, luego se hace lo mismo con $k=2$ y finalmente con $k=1$ [61].

La heurística de Van Breedam [62] propone cuatro operadores (String Cross, String Exchange, String Relocation y String Mix) para el intercambio de nodos

entre un par de rutas. Cada uno de estos operadores realiza un intercambio de nodos entre las rutas, transfiriendo un conjunto de nodos de una ruta a otra, o bien intercambiando un conjunto de nodos entre un par de rutas [63].

7.2.2 METAHEURÍSTICAS

Las Metaheurísticas son algoritmos genéricos de optimización y búsqueda del espacio de soluciones, guiando los procesos de búsqueda para explorar y explotar lo mejor posible el espacio de búsqueda y buscando evitar caer en óptimos locales [64] [65].

Generalmente las metaheurísticas permiten obtener una mejor solución que los métodos heurísticos, debido a la mejor exploración del espacio de soluciones que realizan, teniendo como consecuencia también mayores tiempos de ejecución con respecto a las heurísticas.

Las metaheurísticas se pueden clasificar en Constructivas, Basadas en Trayectorias o Basadas en Poblaciones según [66]:

Las metaheurísticas Constructivas (o Basadas en Trayectorias) son procedimientos que iterativamente, a partir de un análisis, van agregando elementos a la solución (inicialmente vacía). Algunas de las metaheurísticas más comunes que corresponden a esta categoría son GRASP y Optimización basado en Colonias de Hormigas.

Las metaheurísticas de Búsqueda son procedimientos que parten de una solución inicial e iterativamente van guiando las transformaciones o movimientos a realizarse para explorar el espacio de soluciones de manera de ir obteniendo mejores soluciones a través de las iteraciones. Algunas metaheurísticas de búsqueda son Búsqueda Tabú y Búsqueda Local.

Finalmente, las metaheurísticas Evolutivas (o Basadas en Poblaciones) son procedimientos que a partir de un conjunto de soluciones (o población), utilizando determinadas estrategias, iterativamente van evolucionando la población de manera de obtener un mejor conjunto de soluciones. Una de las metaheurísticas más conocidas es la de Algoritmos Genéticos.

A continuación se mencionan algunas metaheurísticas e implementaciones propuestas para la resolución de los problemas VRP, se intenta abarcar al menos una metaheurísticas de cada una de los paradigmas mencionadas anteriormente. Por lo tanto se estudiarán las metaheurísticas de Optimización basado en Colonias de Hormigas y GRASP (correspondientes a las metaheurísticas Constructivas), las metaheurísticas de Búsqueda Tabú y Búsqueda Local (correspondientes a las metaheurísticas de Búsqueda), y la metaheurística de Algoritmos Genéticos (correspondientes a las metaheurísticas Evolutivas)

La Optimización basado en Colonias de Hormigas (ACO, por sus siglas en inglés, Ant Colony Optimization) se inspira directamente en el comportamiento de las colonias reales de hormigas para solucionar problemas de optimización combinatoria. Se basan en una colonia de hormigas artificiales, esto es, agentes computacionales simples, independientes y asíncronos que trabajan de manera cooperativa para encontrar la mejor solución y se comunican mediante modificaciones de ciertos parámetros numéricos (llamados por analogía, rastros de feromonas artificiales) [67] [68].

Los algoritmos de ACO parten de una solución vacía y a medida que se itera en el algoritmo se va creando el grafo solución (algoritmo constructivo); de esta manera, cada hormiga artificial recorre las aristas del grafo (conjunto con los posibles caminos que una hormiga puede tomar) y va depositando un rastro de feromona artificial que indica la “deseabilidad aprendida” del movimiento por la arista [68].

Bell y McMullen [69] proponen una implementación del ACO para resolver el problema CVRP, basándose en algoritmos ACO que resuelven el problema TSP. En la metaheurística desarrollada, cada hormiga se comporta como un vehículo, de manera que cada una intentará visitar a todos los nodos y luego volverá al depósito; si la visita al próximo cliente excede la capacidad del vehículo, se vuelve al depósito desde el último cliente visitado.

El algoritmo lleva una lista con los potenciales clientes a visitar, siendo los clientes que forman esta lista los que se encuentren más cercanos al nodo en el cual se encuentra la hormiga y que a su vez ese cliente esté cercano a otros futuros clientes a visitar (se buscan evitar esfuerzos considerando movimientos a clientes que tienen pocas probabilidades de ser elegido como un camino de la solución); obviamente los clientes que forman esta lista aún no han sido visitados.

Las hormigas, deciden que cliente visitar en función de una fórmula probabilística, la cual considera la “acumulación de feromonas” del camino y la distancia hasta el próximo cliente; a su vez, a medida que se ejecuta el algoritmo se van actualizando las acumulaciones de feromonas en los caminos, aumentando si otra hormiga eligió el mismo camino y disminuyendo si no se ha utilizado (simula la evaporación natural de la feromona y sirve para que un camino no se vuelva demasiado dominante).

Una vez que se obtiene la ruta solución para un vehículo, se busca optimizarlo utilizando la heurística 2-opt.

Este algoritmo es iterativo, obteniéndose una ruta para un vehículo en cada iteración, además se mantiene una lista con los clientes que aún no han sido visitados de manera de intentar visitarlos en la próxima iteración; si no quedan más vehículos o se han atendido todos los clientes, el algoritmo concluye.

Por otra parte, Gambardell et al. [70] proponen un algoritmo para resolver el

problema CVRPTW. Trabajan con dos colonias de hormigas, las cuales cooperan y evolucionan en paralelo (compartiendo información de feromonas); una de las colonias se encarga de minimizar la cantidad de vehículos a utilizar mientras que la otra se encarga de minimizar la distancia necesaria para realizar la distribución.

Ambas colonias trabajan con depósitos duplicados (tantos depósitos como vehículos haya disponibles), de manera que cada depósito duplicado tendrá asignada una ruta, y la solución final será el conjunto de todas las rutas de los depósitos duplicados.

Cada hormiga, al elegir el próximo cliente a visitar, entre otras cosas, chequea que la ventana de tiempo del mismo no haya finalizado.

7.2.2.2 GRASP

La metaheurística GRASP (Greedy Randomized Adaptive Search Procedure) es un procedimiento iterativo, donde cada iteración consiste básicamente en una fase de construcción y una de mejora [71].

En la fase constructiva se va creando iterativamente una solución factible (de buena calidad), agregando un elemento en cada paso de la iteración. A su vez, en cada iteración se elige el próximo elemento a ser agregado a la solución parcial utilizando una función greedy. Esta función selecciona un elemento aleatoriamente de entre un conjunto de elementos, formado por los elementos que brindan mayor beneficio al añadirse a la solución parcial; la elección de estos elementos se realiza sin considerar su efecto en futuras iteraciones.

Las soluciones generadas en esta fase no son necesariamente óptimas, por lo que resulta necesario aplicar un procedimiento de búsqueda local para mejorar la solución obtenida.

El algoritmo de búsqueda local parte de la solución generada en la fase de construcción y sucesivamente va reemplazando la solución actual por otra mejor que se encuentre dentro de la vecindad, finalmente cuando no se encuentra una mejor solución termina.

Gallart [72] propone la implementación de un algoritmo GRASP que permite resolver el problema CVRP. Para determinar el conjunto de clientes candidatos a visitarse se utiliza una función de costo que toma en cuenta la distancia hasta el cliente (desde el nodo actual), la cantidad de vehículos necesarios si se visita ese cliente, y una variable que indica el tráfico existente en la arista que une al nodo actual con el cliente; la función objetivo plantea una relación entre estos tres parámetros, buscando minimizar su valor.

Finalmente, una vez que se obtiene una solución factible, se utiliza una función 2-opt como algoritmo de búsqueda local. A su vez, también se utiliza una metaheurística GRAPS para resolver el problema de embalaje de paquetes en los camiones.

7.2.2.3 BÚSQUEDA TABÚ

La Búsqueda Tabú (Tabú Search) es una metaheurística que en cierta medida guía a los procedimientos de búsqueda local para realizar una mejor exploración del espacio de soluciones, permitiendo escaparse de óptimos locales y continuar la búsqueda hacia soluciones mejores; de esta manera, se realizan búsquedas locales donde se permite optar por soluciones vecinas que aumenten el costo [73].

Para poder llevar esto a cabo, y no volver a una solución de menor costo por la cual ya se pasó, se utiliza una memoria de corto plazo (circular) donde se guardan ciertos atributos de las soluciones visitadas para poder evitar volver a ellas; dichas soluciones “prohibidas” se denominan soluciones tabú (igualmente, y ante ciertas circunstancias, se permiten aceptar soluciones tabú) [74].

Osman [75] plantea una implementación de la metaheurística Tabú Search para resolver el problema CVRP. En el trabajo, se define a la vecindad de una solución como todas las soluciones que se pueden obtener al aplicar el operador λ -intercambio a dicha solución; por lo tanto las vecinas se obtienen de aplicar a lo sumo λ intercambios de clientes entre pares de rutas (la cantidad de clientes intercambiados entre las dos rutas no necesariamente deben ser iguales).

Una vez que se realiza un intercambio entre dos rutas, este “movimiento” queda dentro de los movimientos tabú; a su vez, a las rutas implicadas en el intercambio de clientes se les aplica el algoritmo 2-opt para encontrar el mejor orden para visitar a los clientes en cada ruta.

Gendreau et al. [76] utilizan un algoritmo basado en Búsqueda Tabú para resolver el problema de MDVRP. El mismo parte de una solución inicial (solución generada de manera aleatoria) e iterativamente se van reasignando clientes entre las rutas buscando mejorar la solución. Al momento de reasignar un cliente, se busca insertarlo en una ruta que contenga a uno de los vecinos más cercanos al mismo, al mismo tiempo en cada inserción se ejecuta un algoritmo de optimización local que reordena los clientes de las rutas afectadas buscando mejorarla.

Cada uno de los movimientos necesarios para reasignar clientes queda como un movimiento tabú (por un tiempo); a su vez, se permite considerar rutas (en la reasignaciones de clientes) que no contemplen las restricciones del problema (por ejemplo capacidad) con el objetivo de poder salir de óptimos locales.

7.2.2.4 ALGORITMOS GENÉTICOS

Los algoritmos Genéticos se basan en los procesos de la evolución natural para resolver problemas de búsqueda y optimización.

Se trabaja con un conjunto de individuos (población), cada uno de los cuales representan una solución factible; a su vez, a cada uno de estos individuos se les aplica una función (diseñada para cada problema en particular) que asigna un valor o puntuación a cada uno de ellos que se supone refleja que tan buena o no es una solución.

En cada iteración del algoritmo se modifican los individuos de la población,

creando una nueva “generación”; generalmente en cada iteración se llevan a cabo tres operaciones: la selección, el cruzamiento y la mutación [77].

En la etapa de selección se eligen de manera aleatoria los padres, teniendo mayor probabilidad de ser elegidos aquellos individuos de la población que tengan mayor puntuación (representa una mejor solución).

Una vez seleccionados los padres en la etapa anterior, se lleva a cabo la etapa de cruzamiento. La misma consiste en elegir, en repetidas oportunidades, dos padres pertenecientes conjunto de padres y aplicarles un conjunto de transformaciones a cada uno de esos padres de manera de obtener dos hijos (pertenecientes a la nueva “generación”); generalmente lo que se hace es dividir cada uno de los padres en porciones (denominadas “cromosomas”) e intercambiar dichos genes entre ambos logrando así los hijos (la división de los padres se diseña para cada problema en particular).

Finalmente, se lleva a cabo la etapa de mutación para cada uno de los hijos generados, que consiste en realizar algunas modificaciones en el individuo buscando mejorarlo (se busca mejorar la solución obtenida) [77].

En [78] Pereira et al. plantean un algoritmo genético para resolver el problema CVRP. En el mismo se parte de un conjunto de soluciones sobre las cuales se realizan operaciones de cruce y mutación.

Para cruzar dos soluciones (en una solución existen tantas rutas como vehículos se utilicen), se toma una sub-ruta de una de las rutas pertenecientes a una solución y se la inserta en una ruta de la otra solución. Más específicamente, se toman dos soluciones (s_1 y s_2) y una sub-ruta de una de las rutas pertenecientes a s_1 ($sr_1 = x_1 \dots x_n$); notar que x_1 y x_n son una sub-ruta de una ruta, por lo que no tienen por qué ser iguales. Luego se determina el cliente (z_i) más cercano a x_1 de manera que z_i no aparezca en sr_1 ; sea $r_2 = y_1 \dots z_i \dots y_n$ la ruta perteneciente a la solución s_2 que contiene al nodo z_i . Posteriormente se inserta la sub-ruta sr_1 en la ruta r_2 , resultando $r_2 = y_1 \dots z_i, x_1, \dots, x_n, z_i + 1, \dots y_n$; en caso de que la ruta r_2 no sea factible (por capacidad del vehículo por ejemplo) se particiona la misma en tantas rutas como sean necesarias.

Finalmente se obtienen dos hijos, uno de aplicar la operación anterior (conteniendo a r_2), y otro como una copia exacta de s_1 .

Por otra parte, las operaciones de mutación consisten principalmente en el intercambio de posición de los clientes dentro de una ruta, invirtiendo el orden de algunos clientes en la ruta y cambiando de posiciones sub-rutas completas dentro de la ruta.

8 RELEVAMIENTO DE SOFTWARE CARTOGRÁFICO

8.1 SISTEMAS DE INFORMACIÓN GEOGRÁFICA

En [97], se define un SIG como una colección organizada de hardware, software, datos geográficos y personal, diseñado para capturar, almacenar, actualizar, manipular, analizar y mostrar eficientemente, todas las formas de la información geográficamente referenciada. Además, se define como un sistema de información capaz de guardar y usar datos que describen lugares en la superficie de la tierra.

Se relaciona el concepto de sistema de información geográfica a una aplicación que manipula información que tiene relación con un punto geográfico, es decir, la misma puede ser georeferenciada. Esto es, cada dato tiene asociado un valor que podría ser una coordenada respecto a algún sistema de referencia, que lo ubica geográficamente.

Los SIG tienen características particulares que los diferencian del resto de los sistemas de información. La diferencia clave a destacar es que generalmente se utiliza una base de datos geográfica, que manipula los datos geográficos. Este tipo de base de datos, tiene incorporadas a su motor [109] clases concretas que tienen que ver con el concepto de Geometría e incorpora operaciones espaciales entre Geometrías. Para profundizar sobre el tema consultar las notas del curso de Taller de Sistemas de Información Geográfica en [109].

8.1.1 GVSIG DEKSTOP:

Se define como una aplicación de escritorio. La misma está diseñada para almacenar, analizar y desplegar información geográficamente referenciada con el fin de resolver problemas [96]. Puede ser útil para resolver cuestiones de planificación y de gestión.

Es una herramienta orientada al manejo de información geográfica que se caracteriza por tener una interfaz amigable. Cuenta con una gran cantidad de herramientas para trabajar con información de naturaleza geográfica (herramientas de consulta, creación de mapas, redes, etc.). [96]

Está desarrollado en Java y es compatible con plataformas Linux, Windows y Mac. Esta aplicación pertenece a la comunidad de software libre, con licencia GNU/GPL, lo que permite, además de su libre uso, la posibilidad de modificar código y agregar nuevos módulos para extenderla (existen manuales para desarrolladores).

Otra característica del producto es, según la referencia [96], que está diseñado

para ser fácilmente extensible, esto, sumado a la calidad de software libre permite que se desarrolle una mejora continua de la aplicación.

8.1.2 GOOGLE MAPS

Es una aplicación muy popular utilizada diariamente para todo tipo de usuarios finales.

No forma parte de la comunidad de software libre, por lo tanto no se conoce su código fuente y no se puede modificar. Existen dos tipos de licencia, una gratuita y una empresarial con menos restricciones, en [98] se describen los detalles y la diferencia entre ellas. Se puede utilizar en cualquier PC sin necesidad de instalar programas (se necesita un navegador) ni ocupar lugar en el disco duro.

Provee varias APIs para desarrolladores de software que permiten integrar en múltiples tipos de aplicaciones web los mapas de Google. Es muy sencillo de usar para añadir mapas a sitios web, aplicaciones web o móviles, proporciona una amplia gama de servicios y utilidades para la visualización de datos, manipulación de mapas y direcciones. Google Maps proporciona servicios web como una interfaz para poder solicitar a servicios externos, datos del API de Google Maps y utilizarlos en las aplicaciones integradas al mapa de Google Maps.

Actualmente se pueden encontrar las siguientes APIs:

- Google Maps para Flash
- Google Static Maps
- JavaScript de Google Maps

Se describe a continuación algunas características de la interfaz de JavaScript de Google Maps.

Esta API permite, entre otras cosas, consumir servicios web a través de llamadas Ajax desde código JavaScript. Provee de datos muy útiles como por ejemplo, dadas sus coordenadas, se puede obtener el tiempo estimado entre múltiples puntos geográficos a través de distintas vías de transporte. Permite añadir contenido al mapa a través de una variedad de servicios, que permiten crear aplicaciones robustas en un sitio web.

También provee un servicio que devuelve la distancia entre pares de lugares, librando al programador de lidiar con proyecciones geográficas. Es necesaria una conexión permanente a internet para poder utilizarlo.

Existe una gran variedad de tutoriales claros y en español para poder incorporar un mapa de Google en una aplicación web. Los tutoriales son cortos y concretos permitiendo tener un mapa disponible a través de imágenes satelitales y poder visualizarlo sin mucho esfuerzo.

8.1.3 BING MAPS

Similar al caso de Google Maps, Microsoft, provee una API de Bing Maps para integrar sus mapas en una aplicación web. [106]

Existe una versión básica sin costo, que restringe la cantidad de operaciones diarias y exige el registro y creación de una cuenta del desarrollador para obtener una clave de uso. [105]

8.1.4 ARCGIS

ArcGis [33] es una suite de aplicaciones (producido y comercializado por ESRI) que permite crear, recopilar, analizar y compartir datos, modelos y mapas entre un conjunto de usuarios; actualmente es uno de los sistemas líderes a nivel mundial para crear y utilizar sistemas de información geográfica.

El sistema se encuentra disponible para la plataforma Web, dispositivos móviles (smartphones y tablets) y equipos de escritorio.

Provee un conjunto de APIs gratuitas para desarrolladores, que permite integrar mapas a una aplicación web, proporcionando varias funcionalidades asociadas al manejo y creación de diferentes tipos de datos en el mapa (también existe una versión de pago con mayores funcionalidades, mejor respuesta y mayor cantidad de datos permitidos en las consultas); entre las APIs más conocidas y utilizadas se encuentran las de JavaScript y las de ASP .Net. [107] [108]

Una de las funcionalidades que provee ArcGis en su API y que resultan de gran interés (considerando el proyecto actual) es la de poder especificar calles como “no transitables” en un mapa; de esta manera, cuando el servidor envía la respuesta a una ruta entre dos puntos (ruteo), no aparecerán las calles marcadas como “no transitables” (útil en el contexto de un desastre natural).

Lamentablemente, esta API gratuita solo permite el cálculo de ruteo para Estados Unidos y algunos países europeos, debiendo adquirir una licencia paga para poder utilizar esta funcionalidad por ejemplo para Uruguay.

9 RELEVAMIENTO DE SOLVERS PARA PROBLEMAS DE PROGRAMACIÓN MIP

9.1 PROBLEMAS DE PROGRAMACIÓN MIP

En [99] se describe breve y concretamente lo que es un problema de programación lineal LP. Un problema MIP programación entera mixta es un problema LP con la restricción adicional de que alguna (al menos una) de las variables del modelo, sólo puede tomar un valor entero.

9.2 SOLVERS

En el contexto, se trata a los solvers como programas que utilizan distintos algoritmos para intentar resolver problemas de optimización sujetos a restricciones (en esta investigación interesa el caso de restricciones lineales) con variables enteras y continuas (MIP, Mixed Integer Programming).

9.2.1 GAMS

Es un sistema para programación matemática y optimización. Está diseñado para modelar distintos tipos de problemas de optimización, como por ejemplo, problemas de optimización lineales con variables enteras y continuas. [100]

En [101] se pueden ver ejemplos básicos de su uso. La aplicación está compuesta por un compilador y varios solvers, el usuario puede elegir con cuál de ellos quiere resolver. Está basado en el algoritmo de ramificación y poda (branch and bound) y relajaciones. Se pueden ver alternativas en [102] donde se describe qué métodos o algoritmos de resolución utiliza cada solver.

Lamentablemente el software tiene licencia paga, dependiendo de las características que quiera el usuario. El costo es más bajo si se utiliza para fines académicos, que para fines comerciales.

9.2.2 SCIP

Resuelve problemas de programación lineal, programación entera y otros tipos de problemas de optimización.

Tiene una licencia llamada XXX que permite el uso gratuito del software para fines no comerciales y/o académicos. Se puede también, acceder a su código fuente. En la página web de SCIP [103] destaca la figura 3 que compara desempeño entre solvers comerciales y no comerciales.

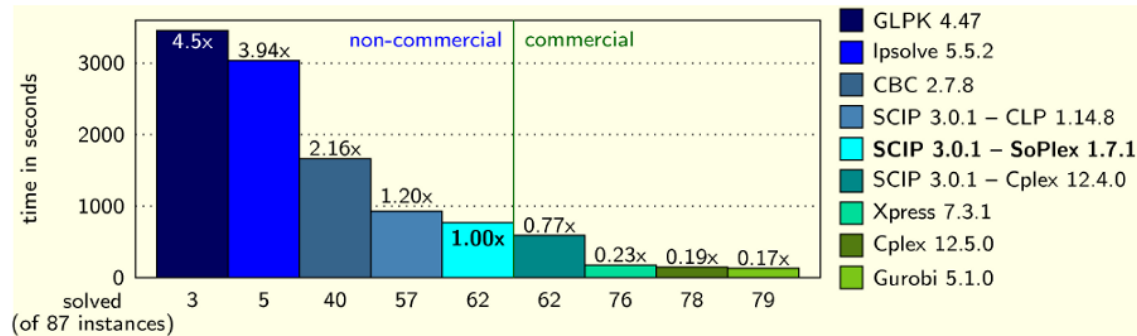


Figura 3

9.2.3 GLPK

Es un paquete escrito en lenguaje C que se utiliza para modelar y resolver problemas de optimización, de programación entera mixta.

En [104] se describe los distintos módulos que lo componen:

- Método simplex primal y dual
- métodos primal y dual simplex
- método de punto interior primal-dual
- método branch-and-cut
- traductor para GNU MathProg
- Interfaz de programación de aplicaciones (API)
- LP / MIP solver

El software tiene licencia GNU GPL por lo que se puede conocer su código fuente, modificarlo y forma parte de la comunidad de software libre. [99]

10 ANEXO

10.1 ANEXO I: SOFTWARE SUMA

Cuando ocurre un desastre significativo, la solidaridad nacional e internacional se moviliza, y en cuestión de horas, comienza a llegar al país afectado una gran cantidad de donaciones por diferentes vías.

Los Administradores de desastres deben estar preparados para recibir gran cantidad de donaciones no solicitadas que pueden no satisfacer las necesidades de la población afectada. Estas donaciones pueden afectar la capacidad local de recepción, almacenaje, clasificación, control y distribución, consumiendo tiempo y recursos al país afectado

El proceso de identificar y clasificar toneladas de suministros y garantizar que los artículos urgentemente requeridos lleguen a las víctimas del desastre pueden abrumar a los trabajadores de socorro [84].

SUMA, el Sistema de Manejo de Suministros Humanitarios de la OPS/OMS, es una herramienta de manejo de información que ayuda a las autoridades nacionales a poner orden en el caos que frecuentemente provoca la asistencia humanitaria mal coordinada [22].

Ayuda a administrar los suministros que llegan y clasificarlos desde su punto de ingreso, con el fin de establecer prioridades en la distribución de los mismos a la comunidad, de acuerdo a las necesidades de la población afectada. Mediante este sistema se puede disponer de información del flujo de las donaciones y del tipo y características de los suministros [22].

Su objetivo ha sido contribuir a resolver los problemas que representa la llegada masiva de asistencia a una región o país afectado por un desastre.

Ayuda a mejorar la administración de la ayuda humanitaria, para que la misma llegue adecuadamente y en tiempo a la población afectada [22].

Es también una herramienta muy útil a la hora de rendir cuentas, garantizar la eficiencia y transparencia en la recepción y distribución de los suministros, permite preparar informes para los donantes, las autoridades nacionales, las agencias humanitarias y la prensa sobre suministros recibidos y entregados a áreas afectadas. La metodología SUMA también puede ser usada en tiempos normales para el manejo del inventario de bodega de por ejemplo hospitales, farmacias, centros de salud [3].

El sistema está conformado por tres niveles [22] [84]:

- Suma Central
- Unidad de Campo
- Módulo de Manejo de Bodegas

Suma Central está diseñado para operar en el lugar donde las autoridades nacionales están administrando el desastre o la emergencia [22] [84].

La unidad de campo ha sido diseñada para trabajar en los puntos de entrada o lugares de recepción, ya sean fronterizos, puertos marítimos o fluviales y grandes centros de recolección locales, a los cuales llegan los suministros durante la emergencia.

Separación e identificación mediante etiquetas de los suministros en las categorías (Urgente, Distribución Inmediata, Distribución no-Urgente, Artículos no Prioritarios).

En el caso de que las computadoras fallen o la necesidad de recolección de datos así lo requiera, se pueden utilizar formularios manuales [22] [84].

El módulo de manejo de bodega es una herramienta que registra las entradas y salidas de suministros, entre otras cosas, en los centros de almacenamiento o bodegas.

Estas bodegas reciben los suministros y la información de los otros niveles del sistema. Esto permite que las instituciones coordinen internamente el manejo de sus suministros, o en coordinación con otras entidades que trabajan en la emergencia, como podría ser el caso de la asignación de un cargamento de medicamentos propiedad de Cruz Roja al Ministerio de Salud [22] [84].

Aparte de estos tres módulos, existe otro modulo, Modulo de Asistencia, que maneja información de suministros antes de su actual llegada al sitio (suministros que están en el “pipeline”). Sirve para hacer seguimiento a las promesas hechas por los donantes y a las necesidades expresadas por el país afectado y luego asignar las promesas a las solicitudes. Este módulo se puede instalar de forma independiente de los otros tres módulos, y es completamente compatible con el software [22] [84].

La idea es que los suministros puedan ser clasificados, inventariados, priorizados y almacenados, desde el mismo punto en el que entran. Todos los donativos sin importar su origen o destinatario deberán ser procesados por SUMA en el punto de entrada antes de ser entregados al destinatario. Esto implica la adopción de políticas y estrategias de coordinación y operación por parte de los organismos e instituciones relacionadas con la administración del socorro, sean gubernamentales o no gubernamentales, antes del desastre.

En cuanto a la protección de la información, la única manera es realizando respaldos regulares. Hay dos opciones para el respaldo y recuperación de los datos: Respaldo de Archivos y Restauración de Archivos, en la versión 5.1 se realiza a través de los famosos y ya obsoletos disquetes.

Actualmente, si bien se sigue utilizando el software SUMA en muchos casos (debido en parte a su fiabilidad y también a que es ampliamente conocido por todos los agentes involucrados en la ayuda humanitaria, sabiendo claramente cómo manejarlo y trabajar con él), también existe otro software que sirve para el manejo eficiente de suministros, denominado LSS (Logistics Support System), el

cual es más moderno y mejora en varios aspectos al SUMA (por ejemplo mejora la coordinación y el intercambio de información entre las partes, brindando la posibilidad además de trabajar sobre la plataforma web), por lo que se le considera como el “sucesor” del SUMA [87].

10.2 ANEXO II: PROGRAMACIÓN POR METAS

La programación por metas, consta en optimizar un conjunto de objetivos (maximizar o minimizar funciones objetivo), cumpliendo con determinadas restricciones y donde el decisor puede otorgar prioridades entre los diferentes objetivos a cumplir. [20]

De esta manera, el decisor, otorgara a cada objetivo un nivel de aspiración que será, o bien el nivel que se quiere alcanzar como mínimo o bien el nivel que no se desea superar, incluso puede ser el nivel que se desea obtener exactamente (igualdad); de esta manera se obtienen las metas que se desean alcanzar. [20]
Por ejemplo, si se desea maximizar con un nivel de aspiración, entonces se busca que ; lo mismo ocurre en el caso de minimizar o de igualdad.

Una vez establecidas las metas, se introducen en las mismas las variables de decisión (n y p), cuya finalidad es representar la diferencia existente entre los niveles de aspiración impuestos por el decisor y los resultados obtenidos en cada función objetivo. [20]

Por ejemplo, si se busca maximizar la función, como se vio anteriormente, se tiene la meta, por lo que al introducir las variables de desviación se obtiene:

y se busca minimizar la variable (variables no deseadas), ya que si entonces y se verifica la meta. Si se está minimizando la función se minimiza y en el caso de igualdad se minimiza. [21]

A la función que minimiza el conjunto de variables no deseadas del problema se le denomina función de realización (f), y en el caso del ejemplo anterior, la función de realización será [21]:

También, el decisor, puede establecer niveles de prioridades entre los diferentes objetivos. Dicha asignación de prioridades puede ser ubicando en cada nivel un solo objetivo, o asignando varios objetivos a un mismo nivel y definir ponderaciones entre los objetivos que comparten nivel (para establecer prioridades dentro del mismo nivel). [20]

Finalmente, la resolución del problema se llevara a cabo tomando en cuenta las metas impuestas y atendiendo las prioridades entre los niveles, de forma que se preferirá por una solución que satisfaga el primer nivel, una vez conseguido esto, se buscara una solución que también verifique el segundo nivel (partiendo de la solución encontrada en el nivel anterior), y así sucesivamente. [20]

Así pues, dados los objetivos, halladas las metas, establecidos los niveles de prioridad y determinadas las funciones de realización, se obtiene el siguiente modelo [21]:

s.a.

En cada nivel j (con $j=1,...,s$) se buscara minimizar la función .

Al conjunto de restricciones originales del problema se les denomina restricciones duras, ya que se deben cumplir necesariamente; al segundo grupo de restricciones, que son las que se generan a partir de las metas, se les denomina restricciones blandas, ya que su cumplimiento no es obligatorio. [21]

A continuación se plantea un pequeño y sencillo ejemplo [21] para ilustrar al lector en cómo se aplica la técnica de programación por metas.

Supongamos que se cuenta con un problema definido de la siguiente manera:

s.a.

El siguiente paso es establecer los niveles de aspiración, construir las metas y determinar las variables no deseadas:

Luego establecemos los niveles de prioridad:

- Nivel 1: Objetivo 1, entonces
- Nivel 2: Objetivo 2, entonces
- Nivel 3: Objetivo 3 y Objetivo 4, entonces

Donde los parámetros “a” y “b” se utilizan para determinar prioridades (ponderar) entre las metas dentro del nivel.

Finalmente, el problema a resolver quedaría de la forma:

s.a.

Lo único que resta ahora es hallar las soluciones para cada nivel. Cabe recordar, que cuando se busca la solución de un determinado nivel, se deben considerar los valores hallados en el nivel anterior y calcular la solución del nivel actual en base a los resultados obtenidos en el nivel anterior (prioridad entre niveles). [20]

11 REFERENCIAS

- [1] Logística y tecnología en la Acción Humanitaria; R. Carrasco, J. Bará, P. Brunet; capítulo 10 del libro “Tecnologías para el Desarrollo de las Comunidades Aisladas”; Ed. Real Academia de Ingeniería; ISBN 978-84-95662-44-6; pp. 427-447; 2011; <http://www.raing.es/sites/default/files/CAP%C3%8DTULO%2010%20-%20CRA.pdf>; último acceso 30/5/2013
- [2] The Applications of Maximal Covering Model in Typhoon Emergency Shelter Location Problem; P. Anping; 2010; 2010 IEEE International Conference; ISSN 2157-3611; DOI 10.1109/IEEM.2010.5674577; pp. 1727 – 1731
- [3] Logística y gestión de suministros humanitarios en el sector salud; Organización Panamericana de la Salud, Organización Mundial de la Salud; 2001; pp. 9-20, 43-51, 55; <http://preparativosyrespuesta.cridlac.org/XML/spa/doc13830/doc13830-contenido.pdf>; último acceso 30/5/2013
- [4] Disaster management and operation research in Uruguay - Reporte Técnico RT 09-08; S. Moscatelli, O. Viera, L. Tansini; 2009; Ed. PEDECIBA Informática - Instituto de Computación (Facultad de Ingeniería - Universidad de la República); ISSN 0797-6410; https://eva.fing.edu.uy/pluginfile.php/25730/mod_resource/content/0/Rep_Tec_Disaster_management_and_Operation_Research_in_Uruguay.pdf; último acceso 30/5/2013
- [5] Sistema de Manejo Integral de Desastres; pp. 1-11, 19-32, 50-51, 64-92; <http://www.medynet.com/usuarios/jraguilar/manejo.pdf>; último acceso 30/5/2013
- [6] Etapas del ciclo de los Desastres; <http://desastres.usac.edu.gt/documentos/pdf/spa/doc12256/doc12256-contenido.pdf>; último acceso 30/5/2013
- [7] Ciclo de los Desastres; Seminario de Capacitación a Comités Distritales – Managua; <http://desastres.usac.edu.gt/documentos/pdf/spa/doc16428/doc16428-1.pdf>; último acceso 30/5/2013
- [8] La Gestión del Riesgo de Desastres: un Enfoque Basado en Procesos; L. Narváez, A. Lavell, G. Pérez; 2009; Proyecto “Apoyo a la Prevención de Desastres en la Comunidad Andina (PREDECAN)”; ISBN 978-9972-787-88-1; Ed. Secretaría General de la Comunidad Andina (Lima - Perú); pp. 9-83; http://www.comunidadandina.org/predecana/doc/libros/PROCESOS_ok.pdf; último acceso 30/5/2013
- [9] Instrumentos de Apoyo para el Análisis y la Gestión de Riesgos Naturales - Guía para el Especialista; COSUDE-Ayuda Humanitaria y Cuerpo Suizo de Ayuda Humanitaria; 2002; pp. 4-44; <http://www.snet.gob.sv/Riesgo/GuiaMetodologica.pdf>; último acceso 30/5/2013
- [10] Gestión del Riesgo de Desastres – Para la Planificación del Desarrollo Local; L. Cosamalón, V. Lovón, M. Sicre, A. Vilela, L. Málaga, G. Masana, A. Montesinos, J. Ruiz; 2009; Proyecto “Fortaleciendo la participación social y las capacidades de gestión de los gobiernos locales en el proceso de reconstrucción”; Ed. Mesa de Concertación para la Lucha contra la Pobreza y Cáritas del Perú; pp. 13-48; <http://bvpad.indec.gov.pe/doc/pdf/esp/doc1369/doc1369.pdf>; último acceso 30/5/2013

30/5/2013

- [11] Guía Municipal para la Gestión del Riesgo; R. Vargas, J. Bárcenas, A. García, D. Peña, G. Coral, H. Gómez, I. Caicedo; 2010; Proyecto de asistencia técnica en gestión local del riesgo a nivel municipal y departamental en Colombia - Programa de Reducción de la Vulnerabilidad Fiscal del Estado frente a Desastres Naturales; pp. 5-44; <http://www.boyaca.gov.co/index.php?idcategoria=18875&download=Y>; último acceso 30/5/2013
- [12] Modelos Matemáticos de Ayuda a la Decisión en Logística Humanitaria; B. Vitoriano, T. Ortuño, G. Tirado, T. Rodríguez, J. Montero; 2012; Seminario "Modelos de Ayuda a la Decisión en Gestión de Desastres y Logística Humanitaria" (Facultad de Ciencias Económicas y Empresariales – Universidad de Granada); <http://www.cdlmadrid.org/cdl/htdocs/universidaddeotono/unioto/matematicas/modelosmatematicos.pdf>; último acceso 30/5/2013
- [13] HELIOS Software; <http://www.humanitarianlogistics.org/helios-software>; último acceso 30/5/2013
- [14] A multi-criteria optimization model for humanitarian aid distribution; B. Vitoriano, T. Ortuño, G. Tirado, T. Rodríguez, J. Montero; 2010; Journal of Global Optimization, 2011, Vol. 51, no. 2; ISSN 1573-2916; DOI 10.1007/s10898-010-9603-z; Ed. Springer US; pp. 189-208; último acceso 30/5/2013
- [15] A lexicographical goal programming based decision support system for logistics of Humanitarian Aid; J. Montero, T. Ortuño, J. Rodríguez, G. Tirado, B. Vitoriano; 2010; TOP, 2011, Vol. 19, no. 2; ISSN 1863-8279; DOI 10.1007/s11750-010-0138-8; Ed. Springer US; pp. 464-479
- [16] The bi-objetive stochastic covering tour problem; F. Tricoire, S. Graf, W. Gutjhar; 2012; Computers & Operations Research, Vol. 39, no. 7; Ed. Elsevier; pp. 1582–1592; <http://www.sciencedirect.com/science/article/pii/S030505481100270X>; último acceso 30/5/2013
- [17] A multi-objetive robust stochastic programming model for disaster relief logistics under uncertainty; A. Bozorgi-Amiri, M. Jabalameli, S. Mirzapour Al-e-Hashem; 2011; OR Spectrum; ISSN 1436-6304; DOI 10.1007/s00291-011-0268-x; Ed. Springer-Verlag; pp. 640-644
- [18] The logistics of bulk relief supplies; R. Knott; 1987; Disasters, vol. 11, no. 2; DOI 10.1111/j.1467-7717.1987.tb00624.x; pp. 113-115
- [19] Multiperiod integrated routing and scheduling of World Food Programme cargo planes in Angola; V. De Angelisa, M. Melcolia, C. Nikoib, G. Storchia; 2007; Computers & Operations Research, Vol. 34, no. 6; Ed. Elsevier; DOI 10.1016/j.cor.2005.07.012; pp. 1601–1615; <http://www.sciencedirect.com/science/article/pii/S030505480500239X>; último acceso 30/5/2013
- [20] Programación por Metas (Goal Programming): pasado, presente y futuro; C. Romero; 2002; Revista Electrónica de Comunicaciones y Trabajos de ASEPUMA; ISBN 9788484426301; Ed. Tirant Lo Blanch; pp. 75-89; http://www.uv.es/sala/malaga/recta/extraordinarios/Vol_01/04t.pdf; último acceso 30/5/2013
- [21] Programación por metas - Programación Matemática para Economistas; R. Caballero, T. Gómez, M. González, M. Hernández, F. Miguel, J. Molina, M. Muñoz, L. Rey, F. Ruiz;

- Curso 2011/2012 “Programación Matemática para Economistas” (Facultad de Ciencias Económicas y Empresariales – Universidad de Málaga); <http://ecomat.ccee.uma.es/mateco/Docencia/PM/Leccion%204/L4P3.pdf>; último acceso 30/5/2013
- [22] Sistema de Manejo de Suministros Humanitarios (SUMA); <http://www.disaster-info.net/SUMA/spanish/index.htm>; último acceso 30/5/2013
- [23] Cómo elaborar un Plan de Evacuación; Dpto. Instrucción Técnica de Prevención – Dirección de Educación Civil - Ministerio de Seguridad Dirección General de Defensa Civil de la Provincia de Buenos Aires; <http://www.mseg.gba.gov.ar/defensacivil/educacivil/plan%20de%20evacuacion.pdf>; último acceso 30/5/2013
- [24] Plan de evacuación; N. Posada; 2001; Capítulo 2 del documento “CBR y toma de decisiones en el contexto de un GIS, caso del volcán Popocatépetl” – Tesis de Grado (Escuela de Ingeniería - Universidad de las Américas Puebla - México); pp. 19-27; http://catarina.udlap.mx/u_dl_a/tales/documentos/msp/posada_t_n/capitulo2.pdf; último acceso 30/5/2013
- [25] Guía para la elaboración - Plan de Emergencias; pp. 8-10; http://www.tramites.go.cr/manual/espanol/10/plan_emergencia.pdf; último acceso 30/5/2013
- [26] Algoritmos Genéticos Aplicados al Diseño de una Red de Comunicaciones Confiable; D. Calegari; 2002; Proyecto de Grado (Facultad de Ingeniería – Universidad de la Republica – Uruguay); pp. 13-19; <http://www.fing.edu.uy/inco/pedeciba/bibliote/tgradoinf/tg-calegari.pdf>; último acceso 30/5/2013
- [27] Resolución del problema de los caminos disjuntos en grafos dirigidos usando técnicas metaheurísticas; B. Marti, A. Sanchez, A. Duarte; 2007; Jornada sobre Algoritmos Evolutivos y Metaheurísticas (CEDI 2007) - Zaragoza (Spain); pp. 163-169; http://gavab.escet.urjc.es/blosxom/publicaciones_new/ano2007/dedp_final.pdf; último acceso 30/5/2013
- [28] Solving a location-routing problem with a multiobjective approach: the design of urban evacuation plans; J. Coutinho-Rodrigues, L. Tralhão, L. Alcáda-Almeida; 2012; Journal of Transport Geography, Vol. 22; DOI 10.1016/j.jtrangeo.2012.01.006; pp. 206–218; <http://www.sciencedirect.com/science/article/pii/S0966692312000105>; último acceso 30/5/2013
- [29] Pattern-based evacuation planning for urban áreas; S. Bretschneider, A. Kimms; 2012; European Journal of Operational Research, Vol. 216, no. 1; DOI 10.1016/j.ejor.2011.07.015; pp. 57–69; <http://www.sciencedirect.com/science/article/pii/S0377221711006175>; último acceso 30/5/2013
- [30] Computer-based model for flood evacuation emergency planning; S. Simonovic, S. Ahmad; 2007; Natural Hazards, Vol. 34, no. 1; ISSN 1573-0840; Ed. Kluwer Academic Publishers; DOI 10.1007/s11069-004-0785-x; pp. 25-51; http://www.academia.edu/936050/Computer-based_model_for_flood_evacuation_emergency_planning; último acceso 30/5/2013
- [31] Optimal Evacuation Scheme Based on Dam-Break Flood Numerical Simulation; W. Xiaoling, S. Ruirui, Z. Zhengyin, H. Ling; 2012; Transactions of Tianjin University, Vol.

17, no. 6; ISSN 1995-8196; DOI 10.1007/s12209-011-1622-4; Ed. Tianjin University; pp. 424-430

- [32] MILS - un innovador e intuitivo sistema de evacuación de incendios; <http://revision.bligoo.com/content/view/293450/MILS-un-innovador-e-intuitivo-sistema-de-evacuacion-de-incendios.html#.UC0WnqNjfkz>; último acceso 30/5/2013
- [33] ArcGIS; <http://www.arcgis.com/about/features.html>; último acceso 30/5/2013
- [34] SimWalk; <http://www.simwalk.com/es/>; último acceso 30/5/2013
- [35] Logistics, Disaster Management and Training Program; R. Stephenson; 1993
- [36] A Multiobjective Approach to Locate Emergency Shelters and Identify Evacuation Routes in Urban Areas; L. Almeida, L. Tralhao, L. Santos, J. Couthino; 2008; Journal: Geographical Analysis - GEOGR ANAL, vol. 41, no. 1; DOI 10.1111/j.1538-4632.2009.00745.x; pp. 9-29
- [37] Albergue Temporal; B. Dibaba; Capítulo 8 del documento “Atención y evaluación de la situación de salud de entidades seleccionadas en refugiados de Somalia y Sudán en Etiopía, 1999-2005” – Tesis Doctor en Ciencias Médicas (Universidad de La Habana - Cuba); 2010; pp. 75 – 84; http://www.sld.cu/galerias/pdf/sitios/urgencia/8albergue_temporal.pdf; último acceso 30/5/2013
- [38] Manual Nacional para el manejo de Albergues Temporales; Sociedad Nacional de La Cruz Roja Colombiana - Dirección General del Socorro Nacional; 2008; pp. 9-97; http://www.cruzrojacolombiana.org/publicaciones/pdf/manual_final_albergues_temporales_1912011_035711.pdf; último acceso 30/5/2013
- [39] Plan de Prevención y Respuesta; COPECO Honduras, Proyecto de Bosque y Productividad Rural (PBPR), Fundemun; 2007; pp. 16-19, 25-28; <http://cidbimena.desastres.hn/RIDH/pdf/doch0062/pdf/doch0062.pdf>; último acceso 30/5/2013
- [40] Guía para el Establecimiento de Refugios Temporales; Dirección Nacional de Protección Civil del Estado de San Luis de Potosí; pp. 4-19; [http://201.144.123.34/webproteccion.nsf/001162fdcf442a3a0625770500632c41/\\$FILE/ATTV90UY/Manual%20de%20Refugios%20Temporales.pdf](http://201.144.123.34/webproteccion.nsf/001162fdcf442a3a0625770500632c41/$FILE/ATTV90UY/Manual%20de%20Refugios%20Temporales.pdf); último acceso 30/5/2013
- [41] Software Sahana Eden; <http://sahanafoundation.org/products/eden/>; último acceso 30/5/2013
- [42] A note on the lifted Miller–Tucker–Zemlin subtour elimination constraints for the capacitated vehicle routing problem; K. Imdat, L. Gilbert, B. Tolga; 2003; European Journal of Operational Research, Vol. 158, no. 3; DOI 10.1016/S0377-2217(03)00377-1; Ed. Elsevier; pp. 793–795; <http://www.sciencedirect.com/science/article/pii/S0377221703003771>; último acceso 30/5/2013
- [43] Modelos y Optimización - Heurísticas y Problemas Combinatorios; S. Ramos; 2007; curso Modelos y Optimización I (Facultad de Ingeniería – Universidad de Buenos Aires); pp. 6-16; <http://materias.fi.uba.ar/7114/Docs/ApunteHeurísticas.pdf>; último acceso 30/5/2013
- [44] Scheduling of vehicles from a central depot to a number of delivery points; G. Clarke,

- W. Wright; 1962; Operations Research, Vol. 12, no. 4; DOI 10.1287/opre.12.4.568; pp. 568-581;
<http://read.pudn.com/downloads160/doc/fileformat/721736/Scheduling%20of%20vehicles%20from%20a%20central%20depot%20to%20a%20number%20of%20delivery%20points.pdf>; último acceso 30/5/2013
- [45] Métodos Exactos y Heurísticos para resolver el Problema del Agente Viajero y el Problema de Ruteo de Vehículos; F. Sandoya; 2007; Decimocuarta Jornada en Estadística e Informática – Ecuador;
http://www.icm.espol.edu.ec/jornadas/14/archivos/Diapositivas/SandoyaFernando/conferencia/SandoyaFernando_M%C3%A9todos_exactos_y_heur%C3%ADsticos_para_el%20VRP_jornadas.pdf; último acceso 30/5/2013
- [46] Algorithms for the vehicle routing and scheduling problem with time window constraints; M. Solomon; 1987; Operations Research, Vol. 35, no. 2; DOI 10.1287/opre.35.2.254; pp. 254-265;
http://www.iro.umontreal.ca/~dift6751/paper_solomon.pdf; último acceso 30/5/2013
- [47] The fleet size and mix vehicle routing problem; B. Golden, A. Assad, L. Levy, F. Ghysens; 1984; Computers & Operations Research, Vol. 11, no. 1; DOI 10.1016/0305-0548(84)90007-8; pp. 49–66;
<http://www.sciencedirect.com/science/article/pii/0305054884900078>; último acceso 30/5/2013
- [48] Estudio comparativo de algoritmos heurísticos para el problema del viajante de comercio; S. Rincón, A. Oller, D. Martínez; 2002; Sistemas Informáticos (Facultad de Informática - Curso 2002-03) - Universidad Complutense de Madrid; pp. 22-58;
http://eprints.ucm.es/9021/1/memoria_proyecto_tsp.PDF; último acceso 30/5/2013
- [49] A sequential route-building algorithm employing a generalised savings criterion; R. Mole, S. Jameson; 1976; Journal of the Operational Research Society, Vol. 27, no. 2; DOI 10.1057/jors.1976.95; Ed. Palgrave Macmillan Journals; pp. 503-511
- [50] A parallel route building algorithm for the vehicle routing and scheduling problem with time windows; J. Schulze, T. Fahle; 1997; European Journal of Operational Research, Vol. 66, no. 3; pp. 331–340; Ed. Elsevier;
<https://www.google.com.uy/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CD0QFjAB&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload%3Fdoi%3D10.1.1.28.9751%26rep%3Drep1%26type%3Dpdf&ei=ryiiUeSIBlw8QTY9YDwBg&usq=AFQjCNGraIJCHQ1j9HYXuh1piwYEdlOKLw&sig2=8MWxEuyLBxBkhlkSZvXGbA>; último acceso 30/5/2013
- [51] Desarrollo de un framework para el problema de ruteo de vehículos; M. Vásquez; 2007; Tesis Magister en Gestión de Operaciones – Departamento de Ingeniería Industrial – Facultad de Ciencias Físicas y Matemáticas – Universidad de Chile; pp. 83-105;
http://www.tesis.uchile.cl/tesis/uchile/2007/vasquez_mm/sources/vasquez_mm.pdf; último acceso 30/5/2013
- [52] A heuristic algorithm for the vehicle-dispatch problem; B. Gillett, L. Miller; 1974; Operations Research, Vol. 22, no. 2; DOI 10.1287/opre.22.2.340; Ed. INFORMS; pp. 340-349
- [53] A generalized assignment heuristic for the vehicle routing; M. Fisher, R. Jaikumar; 1981; Networks, Vol. 11, no. 2; DOI 10.1002/net.3230110205; Ed. Wiley Periodicals; pp. 109–124

- [54] Heurística para la Generación de un Conjunto de Referencia de Soluciones que Resuelvan el Problema de Ruteo de Vehículos con Múltiples Depósitos MDVRP; J. López, S Nieto; 2012; Tenth LACCEI Latin American and Caribbean Conference (LACCEI 2012) - Building Infrastructure by fostering engineering collaboration, efficient and effective integration and innovative planning (2012 - ciudad de Panama); <http://www.laccei.org/LACCEI2012-Panama/RefereedPapers/RP029.pdf>; último acceso 30/5/2013
- [55] Solution To Multi-Depot Vehicle Routing Problem Using Genetic Algorithms; P. Surekha, S. Sumathi; 2011; World Applied Programming, Vol. 1, no 3; pp. 118-131; <https://www.google.com.uy/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CD oQFjAB&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload%3Fdoi%3D10.1.1.207.9651%26rep%3Drep1%26type%3Dpdf&ei=08WkUeOvIZPK9QS604D4AQ &usg=AFQjCNGhvhdAJlXcBK8AibhaT8C6WRDUaw&sig2=F1itPXyGP31wlfCUgCijvA>; último acceso 30/5/2013
- [56] Route first - Cluster second methods for vehicle routing; J. Beasley; 1983; Omega, Vol. 11, no. 4; pp. 403-408; Ed. Elsevier Ltd; http://people.brunel.ac.uk/~mastjjb/jeb/papers/omega_83.pdf; último acceso 30/5/2013
- [57] Heurísticas para Problemas de Ruteo de Vehículos; A. Olivera; 2004; Instituto de Computación, Facultad de Ingeniería, Universidad de la Republica, Uruguay; pp. 21-27; <http://www.fing.edu.uy/inco/pedeciba/bibliote/reptec/TR0408.pdf>; último acceso 30/5/2013
- [58] Computer solutions of the traveling salesman problem; S. Lin; 1965; Bell System Technical Journal, Vol. 44; pp. 2245-2269; <http://www3.alcatel-lucent.com/bstj/vol44-1965/articles/bstj44-10-2245.pdf>; último acceso 30/5/2013
- [59] Adaptación de la heurística de Lin-Kerningham para la resolución de problemas de itinerarios; J. Bautista, J. Pereira; 2003; Actas del XXVII Congreso Nacional de Estadística e Investigación Operativa (2003 – Lleida, España); ISBN 84-8409-955-5; pp. 3013-3027; http://web.udl.es/usuaris/esi2009/treballs/31_1_1.pdf; último acceso 30/5/2013
- [60] An effective heuristic algorithm for the traveling salesman problem; S. Lin, B. Kernighan; 1973; Operations Research. vol. 21 no. 2; DOI 10.1287/opre.21.2.498; pp. 498-516; <http://www.docstoc.com/docs/86700704/An-Effective-Heuristic-Algorithm-for-the-Travelling-Salesman-Problem>; último acceso 30/5/2013
- [61] Improvements to the Or-opt Heuristic for the Symmetric Traveling Salesman Problem; G. Babin, S Deneault, G. Laporte; 2005; Cahiers du GERAD no. G-2005-02; https://www.google.com.uy/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CC 4QFjAA&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload%3Fdoi%3D10.1.1.89.9953%26rep%3Drep1%26type%3Dpdf&ei=OkqiUeOWElmq9AS7koAQ&usg=AFQjCNEimiy6_PzOwxiocj-9AF3VVWwQmPw&sig2=Fjfvsaoh907_I57cpkiWqw; último acceso 30/5/2013
- [62] Improvement heuristics for the Vehicle Routing Problem based on simulated annealing; A. Van Breedam; 1995; European Journal of Operational Research, Vol. 86, no. 3; DOI 10.1016/0377-2217(94)00064-J; Ed. Elsevier; pp. 480-490; <http://www.sciencedirect.com/science/article/pii/037722179400064J>; último acceso 30/5/2013

- [63] Multi-Route Improvement Algorithm; <http://neo.lcc.uma.es/vrp/solution-methods/heuristics/multi-route-improvement-algorithm/>; último acceso 30/5/2013
- [64] Metaheuristics; S. Ólafsson; 2006; Handbook on Simulation, Handbooks in Operations Research and Management Science, Vol. 7, pp. 633-654; ISBN 978-0-444-51428-8; Ed. Elsevier; pp. 1-9, 13-16; <http://www.public.iastate.edu/~olafsson/metaheuristics.pdf>; último acceso 30/5/2013
- [65] Introducción a los Algoritmos Metaheurísticos; F. Herrera; 2006; Seminario de Sistemas Complejos, Algoritmos Evolutivos y Bioinspirados (Universidad Internacional de Andalucía); <http://sci2s.ugr.es/seminars/5/Int-Metaheuristics-UIA-Baeza-2006.pdf>; último acceso 30/5/2013
- [66] Metaheurísticas: Concepto y Propiedades; J. Moreno; 2004; Departamento de Estadística, I.O. y Computación - Universidad de la Laguna – Santa Cruz de Tenerife; <http://www.gi.ulpgc.es/tebadm/almacen/seminarios/MH%20Las%20Palmas%202.pdf>; último acceso 30/5/2013
- [67] Algoritmos de Optimización basados en Colonias de Hormigas aplicados al Problema de Asignación Cuadrática y otros problemas relacionados; F. Arito; 2010; Tesis Licenciatura en Ciencias de la Computación (Facultad de Ciencias Físicas, Matemáticas y Naturales - Universidad Nacional de San Luis – Argentina); pp. 1-32; https://www.google.com.uy/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CC4QFjAA&url=http%3A%2F%2Fwww0.unsl.edu.ar%2F~legui%2Fdocuments%2Futils.php%3Faction%3Ddownload%26filename%3Dtesis-f.pdf&ei=3lWiUcaUEpGO9ATFn4GgBg&usg=AFQjCNH1oWnGb1ZKS8CL69_Ox3sqvNWT rQ&sig2=uDblyN8aUKqJh8f38aVnYg; último acceso 30/5/2013
- [68] La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques; S. Alonso, O. Cordón, I. Fernández, F. Herrera; 2004; Optimización inteligente: técnicas de inteligencia computacional para optimización, pp. 261-314; ISBN 84-9747-034-6; Ed. Universidad de Málaga (UMA); pp. 1-22; [http://sci2s.ugr.es/publications/ficheros/OCH%20Modelos%20y%20Nuevos%20Enfoques%20\(Chapter\).pdf](http://sci2s.ugr.es/publications/ficheros/OCH%20Modelos%20y%20Nuevos%20Enfoques%20(Chapter).pdf); último acceso 30/5/2013
- [69] Ant colony optimization techniques for the vehicle routing problem; J. Bell, P. McMullen; 2004; Advanced Engineering Informatics, Vol. 18, no. 1; DOI 10.1016/j.aei.2004.07.001; Ed. Elsevier; pp. 41-48; <http://www.sciencedirect.com/science/article/pii/S1474034604000060>; último acceso 30/5/2013
- [70] MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows; L. Gambardella, E. Taillard, G. Agazzi; 1999; Technical Report IDSIA; Ed. London; pp. 63-76; <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.45.5381&rep=rep1&type=pdf>; último acceso 30/5/2013
- [71] Greedy randomized adaptive search procedures; L. Pitsoulis, M. Resende; 2002; Handbook of Applied Optimization; ISBN 978-0195125948; pp. 168-181; Ed. Oxford University Press; pp. 1-16; <http://www2.research.att.com/~mgcr/doc/grasp-hao.pdf>; último acceso 30/5/2013
- [72] Análisis, diseño e implementación de un algoritmo meta heurístico GRASP que permita resolver el problema de rutas de vehículos con capacidad; J. Gallart; 2009; pp. 1-27, 51-58; Tesis para Título de Ingeniero Informático (Universidad Católica del Perú);

http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/331/GALLART_JOSEPH_AN%C3%81LISIS_DISE%C3%91O_E_IMPLEMENTACI%C3%93N_DE_UN_ALGORITMO_METAHEURISTICO_GRASP_QUE_PERMITA_RESOLVER_EL_PROBLEMA_DE_RUTAS_DE_VEHICULOS_CON_CAPACIDAD.pdf?sequence=1; último acceso 30/5/2013

- [73] A Tutorial on Tabu Search; A. Hertz, E. Taillard, D. de Werra; 2011; Curso G5BAIM en "School of Computer Science and IT" (Universidad de Nottingham – Reino Unido); <http://red.cs.nott.ac.uk/~ajp/courses/g5baim/files/IntroTS.pdf>; último acceso 30/5/2013
- [74] Introducción a la Búsqueda Tabú; B. Melián, F. Glover; 2004; Procedimientos Metaheurísticos en Economía y Empresa; pp. 29 – 71; Ed. Tirant lo Blanch; ISBN 8484569381; [http://leeds-faculty.colorado.edu/glover/ts%20-%20intro%20-%20metaheuristics%20in%20econ%20&%20bus%20-%20w%20belen%20%20\(spanish\).pdf](http://leeds-faculty.colorado.edu/glover/ts%20-%20intro%20-%20metaheuristics%20in%20econ%20&%20bus%20-%20w%20belen%20%20(spanish).pdf); último acceso 30/5/2013
- [75] Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problema; I. Osman; 1993; Annals of Operations Research, Vol. 41, no. 4; pp. 421-451
- [76] A Tabu Search Heuristic for the Vehicle Routing Problem; M. Gendreau, A Hertz, G. Laporte; 1994; Operations Research/Computer Science Interfaces Series, Vol. 30; 2005; Ed. Springer US; ISBN 978-0-387-23667-4; pp. 145-163; pp. 2-9; <http://www.cs.amherst.edu/~ccm/cs34/papers/tabuveh2661622.pdf>; último acceso 30/5/2013
- [77] Genetic Algorithms: An Overview; M. Mitchell; 1995; An Introduction to Genetic Algorithms, Vol. 1; Ed. MIT Press; ISBN 0262631857; pp. 31-39; http://ecce.admu.edu.ph/wiki/pub/Main/ResearchProjects/mitchell_GA_tutorial.pdf; último acceso 30/5/2013
- [78] GVR: a New Genetic Representation for the Vehicle Routing Problem; F. Pereira, J. Tavares, P. Machado, E. Costa; 2002; AICS 2002 Proceedings of the 13th Irish International Conference on Artificial Intelligence and Cognitive Science; Ed. Springer-Verlag; ISBN 3-540-44184-0; pp. 95-102; http://neo.lcc.uma.es/radi-aeb/WebVRP/data/articles/vrp_aics2002.pdf; último acceso 30/5/2013
- [79] Control de Inventarios Investigación de Operaciones 4; M.J. Moya Navarro pp. 19-25; http://books.google.es/books?id=uG8_nuimuhAC&printsec=frontcover&hl=es#v=onepage&q&f=false
- [80] Introduction to Logistics Systems Planning and Control; G. Ghiani, G. Laporte and R. Musmanno; 2004; paginas 14-18, 103-137; ISBN 9780470014042; DOI 10.1002/0470014040; Eds. S. Ross, R. Weber; http://www.pc-freak.net/international_university_college_files/Introduction%20to%20Logistic%20Systems%20Planning%20&%20control.pdf; último acceso 30/5/2013
- [81] GESTIÓN DE STOCKS: MODELOS DETERMINISTAS; Ángel A. Juan, Rafael García Martín; <http://ww.investigacion-operaciones.com/material%20didactico/Modelo%20Inventarios%201.pdf>; último acceso 30/5/2013
- [82] Modelos de Optimización en Operaciones de Logística Humanitaria durante inundaciones en Villahermosa, Tabasco; C. Mejía, O. Rodríguez, J. Gaytán; 2012; XXII Congreso Nacional de Hidraulica (Acapulco, Guerrero, Mexico); http://www.revistatlaloc.org.mx/amh_congreso/articulos/CambioClimaticoyEventosEx

- tremos/026ART_mac%5B1%5D.pdf; último acceso 30/5/2013
- [83] Sahana Eden; <http://sahanafoundation.org/products/eden/>; último acceso 30/5/2013
- [84] SUMA - MANUAL PARA EL USUARIO Organización Panamericana de la Salud; 525 Twenty-third, N.W. Washington D.C. 20037, EUA; <http://www.disaster-info.net/SUMA/spanish/index.htm>; último acceso 30/5/2013
- [85] A Stochastic Humanitarian Inventory Control Model for Disaster Planning
Autores: K. Ozbay, E. Ozguven; 2006; Transportation Research Record: Journal of the Transportation Research Board, No. 2022; ISSN: 0361-1981; pp. 63-75; <http://www.rits.rutgers.edu/files/disasterplanning.pdf>; último acceso 30/5/2013
- [86] Humanitarian Relief Chains: Issues and Challenges. BEAMON, B.M. En Proceedings of the 34th International Conference on Computers and Industrial Engineering, Vol. 34, pp. 77-82, 2004; <http://www.docstoc.com/docs/31881727/HUMANITARIAN-RELIEF-CHAINS>, último acceso 30/5/2013
- [87] Logistics Support System (LSS); diapositivas 1-30; <http://www.itu.int/ITU-D/emergencytelecoms/events/Alexandriaconference/presentations/Doc41-Fundesuma.pdf>; último acceso 30/5/2013
- [88] HADS, a goal programming-based humanitarian aid distribution system; J. Montero, T. Ortuño, J. Rodríguez, G. Tirado, B. Vitoriano; 2010; Journal of Multi-Criteria Decision Analysis, Vol. 16, no. 2; DOI 10.1002/mcda.439; pp. 55-64; Ed. John Wiley & Sons
- [89] Buddy Guard;
http://www.disaster.net/apps/details/buddyguard_emergency_group_sms; último acceso 30/5/2013
- [90] Pocket First Aid & CPR;
http://www.heart.org/HEARTORG/CPRAndECC/CommunityTraining/CommunityProducts/Apps---Pocket-First-Aid-CPR_UCM_308819_Article.jsp; último acceso 30/5/2013
- [91] Top Apps for Natural Disasters;
http://articles.software.informer.com/top_apps_for_natural_disasters.html; último acceso 30/5/2013
- [92] Cruz Rojo Boliviana;
http://www.cruzrojaboliviana.org/dndo/publicaciones/revista/7_19/articulo_15.htm; último acceso 30/5/2013
- [93] Capacity Building in Asia using Information Technology Applications; Disaster Managment, module 4; <http://www.adpc.net/casita/course-materials/Mod-4-Disaster-Mgmt.pdf>; último acceso 30/5/2013
- [94] SINAIE - Ley Nº18621;
<http://www.parlamento.gub.uy/leyes/AccesoTextoLey.asp?Ley=18621&Anchor=>; último acceso 30/5/2013
- [95] SINAIE - Catálogo Histórico de Desastres del Uruguay;
http://www.sne.gub.uy/index.php?option=com_content&view=article&id=230&Itemid=87; último acceso 30/5/2013
- [96] GVSIG; <http://www.gvsig.org/web/home/projects/gvsig-desktop>; último acceso

30/5/2013

- [97] ESRI Understanding GIS--The Arc Info method; <http://www.ciesin.org/docs/005-331/005-331.html>; último acceso 30/5/2013
- [98] Licencia del API de Google Maps; <https://developers.google.com/maps/licensing?hl=es->; último acceso: 30/5/2013
- [99] Reference Manual for GLPK Version 4.45; Diciembre 2010 paginas 12-13, pp 220-228; <http://kam.mff.cuni.cz/~elias/optimed/glpk.pdf>; último acceso: 30/5/2013
- [100] The GAMS System; <http://www.gams.com/docs/intro.htm>; último acceso: 30/5/2013
- [101] GAMS Example; <http://www.gams.com/docs/example.htm>; último acceso 30/5/2013
- [102] GAMS Solvers; <http://www.gams.com/solvers/index.htm>; último acceso 30/5/2013
- [103] SCIP - Solving Constraint Integer Programs; <http://scip.zib.de>; último acceso 30/5/2013
- [104] Página de GLPK (GNU Linear Programming Kit); <http://www.gnu.org/software/glpk/>; último acceso 30/5/2013
- [105] Getting a Bing Maps Key; <http://msdn.microsoft.com/en-us/library/ff428642.aspx#bmtou>; último acceso 30/5/2013
- [106] Bing Maps Developer Resources Map SDKs, APIs, Tips, Training; <http://www.microsoft.com/maps/developers/web.aspx>; último acceso 30/5/2013
- [107] ArcGis API Javascript; http://resources.arcgis.com/es/content/enterprisegis/10.0/web_app_javascript
- [108] ArcGis API ASP .Net; http://resources.arcgis.com/es/content/enterprisegis/10.0/web_app_dotnet
- [109] Teórico de Bases de Datos Geográficas, curso de Taller de Sistemas de Información Geográficos Empresariales InCo Facultad de Ingeniería UDELAR; Diapositivas 20-42; Web <https://eva.fing.edu.uy/>; último acceso 30/5/2013

MANUAL DE USUARIO

En este documento se describe el uso de las funcionalidades básicas del prototipo de software desarrollado.

Introducción

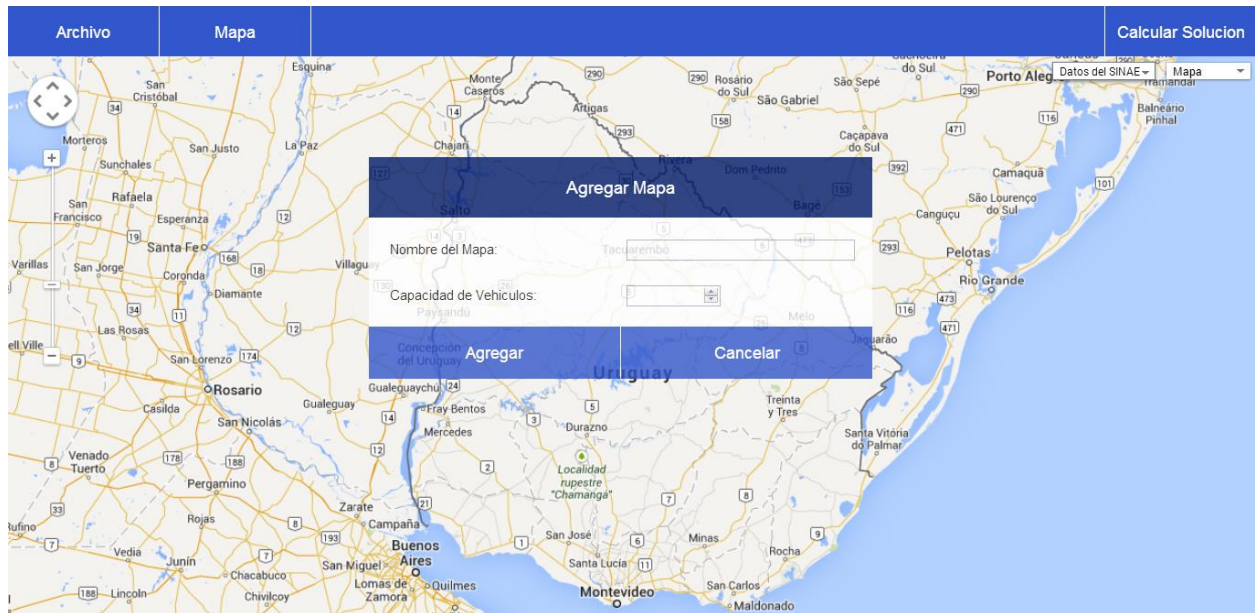
Este sistema permite resolver el problema de distribución de suministros y o personas desde ciertos puntos de partida hacia donde sean necesitados, buscando minimizar los tiempos necesarios para llevar la distribución a cabo, determinando cuáles destinos se visitan desde qué orígenes, en qué orden y cuántas unidades de suministros es transportado en cada tramo y por consecuencia, cuántas unidades se dejan en cada sitio.

Se define un conjunto de proveedores y un conjunto de demandantes que representan el lugar geográfico en el que hay personas que necesitan suministros. Puede llegar a ser muy complicado decidir cómo y en qué orden visitar a las zonas necesitadas. El sistema resuelve en poco tiempo de ejecución, la forma de distribuir todos los suministros y garantizar que la distribución sea aproximadamente óptima en términos del tiempo transcurrido desde que los vehículos salen del origen hasta que llegan a la zona afectada.

El sistema también brinda las clásicas funcionalidades de un mapa web como cambiar mapa de fondo, zoom in, zoom out, etc.

Crear un mapa nuevo

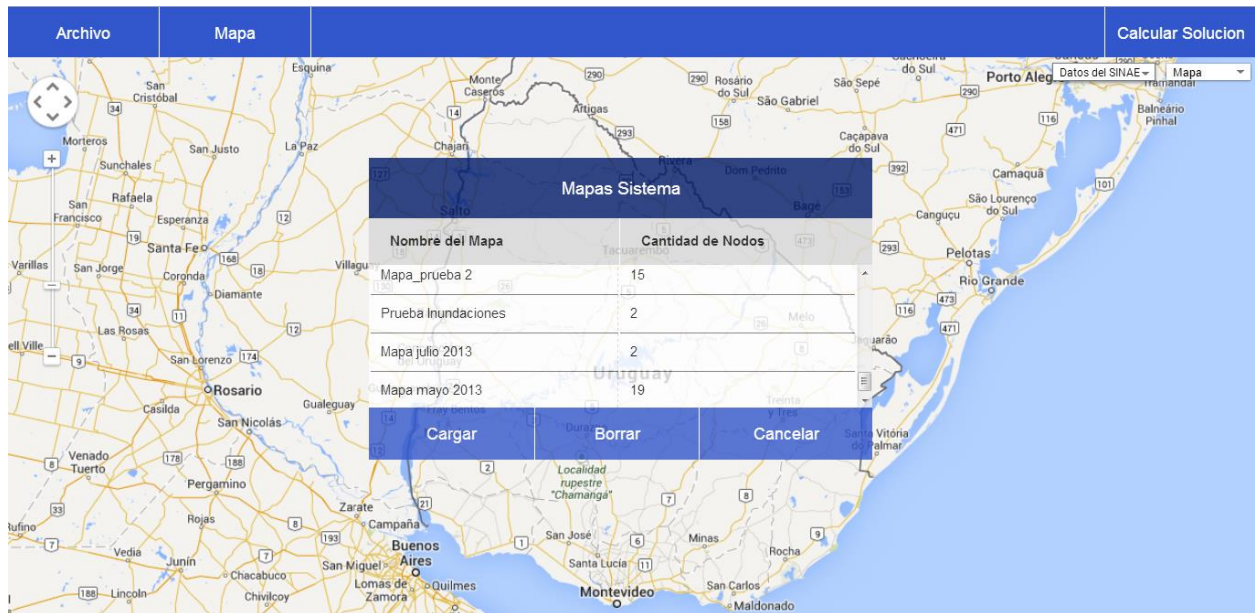
Para crear un mapa nuevo, seleccionar la opción 'Nuevo' en el menú 'Archivo' de la barra superior izquierda de la pantalla. A continuación, el sistema muestra un cuadro de diálogo similar al de la siguiente imagen.



Debe completar los campos con los valores correspondientes. Para crear el mapa seleccionar 'Aceptar' o 'Cancelar' para descartar los cambios.

Cargar un mapa guardado

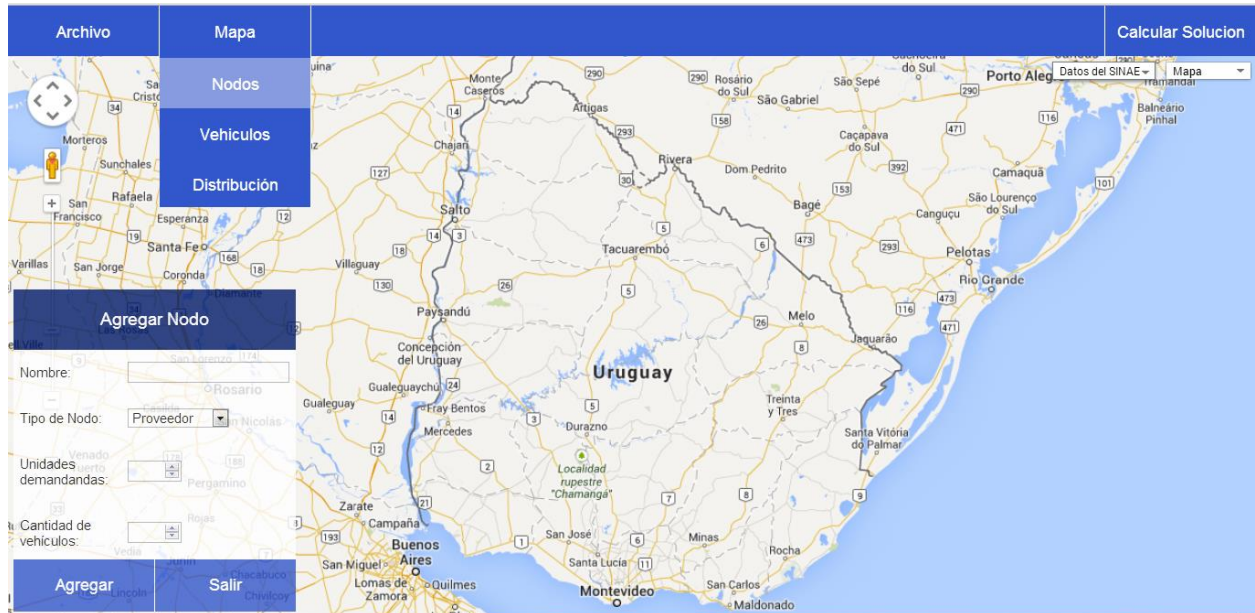
Para cargar un mapa, seleccionar la opción 'Cargar' en el menú 'Archivo' de la barra superior izquierda de la pantalla. A continuación, el sistema muestra un cuadro de diálogo con la lista de mapas guardados en el sistema similar al de la siguiente imagen.



Seleccionar el mapa deseado y luego seleccionar la opción Aceptar para cargar el mapa o seleccione Cancelar para descartar.

Agregar un nodo (demandante o proveedor)

Para agregar un nodo al mapa actual, seleccionar la opción Agregar en el menú Mapa de la barra superior izquierda de la pantalla. A continuación, se despliega un cuadro de diálogo similar al de la siguiente imagen.

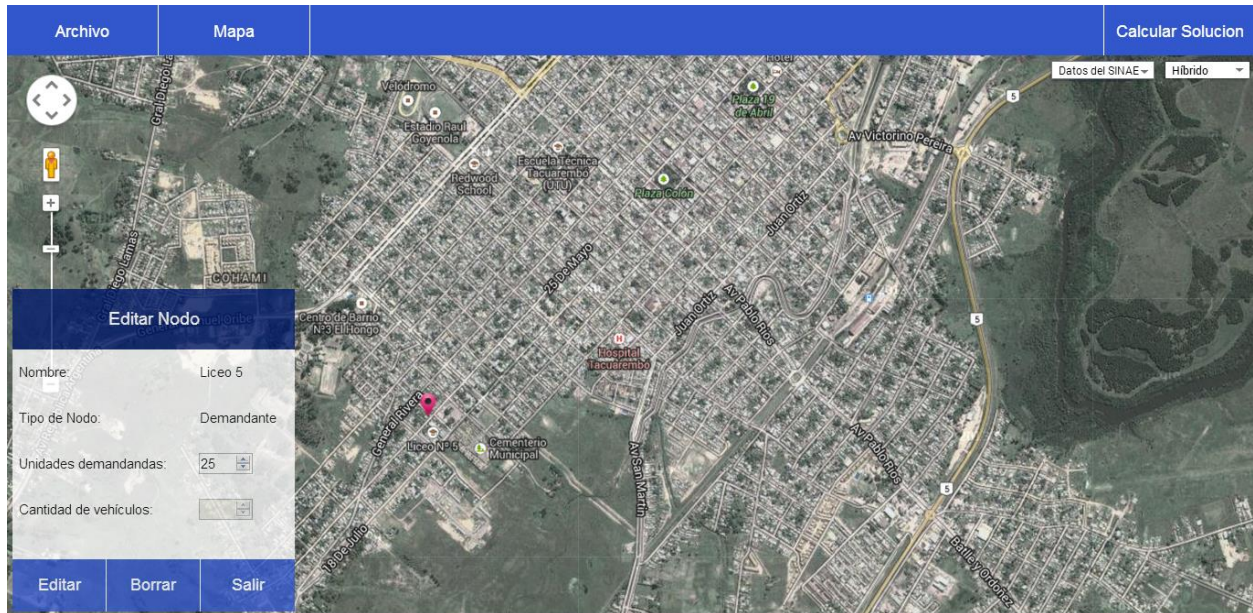


Debe llenar los campos con los valores correspondientes. Por defecto, se crea un nodo de tipo proveedor, si se desea crear un nodo de tipo demandante, se debe cambiar el tipo de nodo.

Tanto para crear nodos de tipo proveedor como demandante, luego de completar los valores correspondientes debe seleccionar 'Aceptar', inmediatamente después de este paso, debe seleccionar una ubicación en el mapa para el nodo. Para elegir la ubicación, simplemente se hace click en el lugar deseado del mapa. Si se selecciona 'Cancelar' se descartan los cambios. Se pueden agregar como máximo 15 nodos.

Modificar un nodo

Se selecciona un nodo en el mapa, a continuación se despliega un cuadro de diálogo que muestra los valores del mismo.



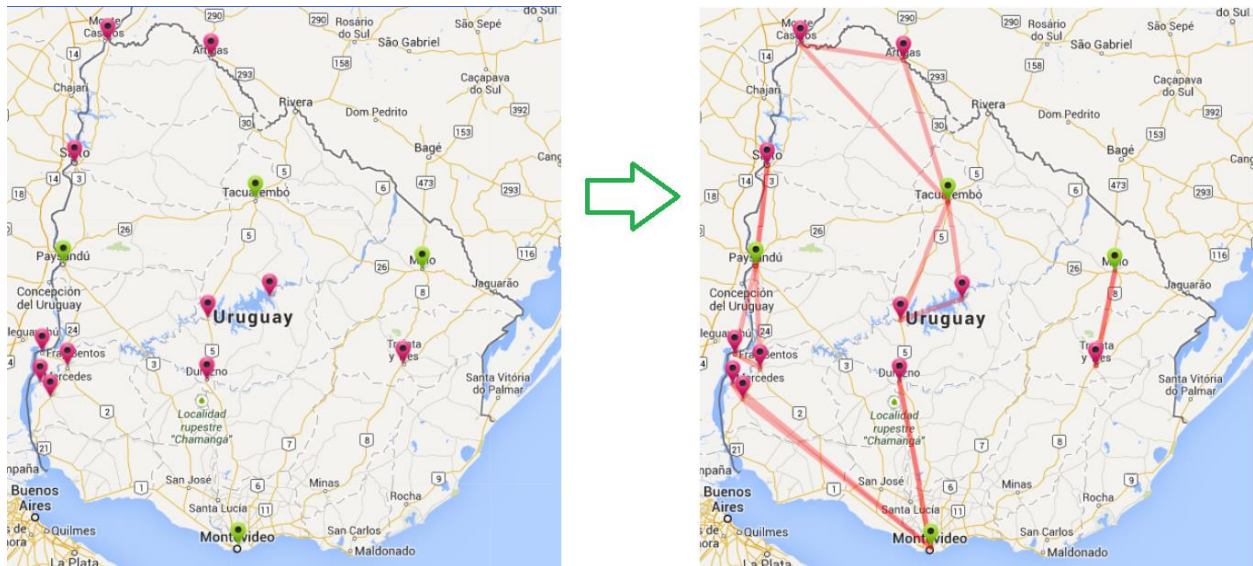
Luego de cambiar uno o más valores se selecciona 'Editar' para guardar los cambios o 'Salir' para descartarlos.

Eliminar un nodo

Se selecciona un nodo en el mapa, a continuación se despliega un cuadro de diálogo que muestra los valores del mismo, como en el caso de Modificar un nodo. Si se selecciona la opción 'Borrar', el nodo es eliminado del mapa.

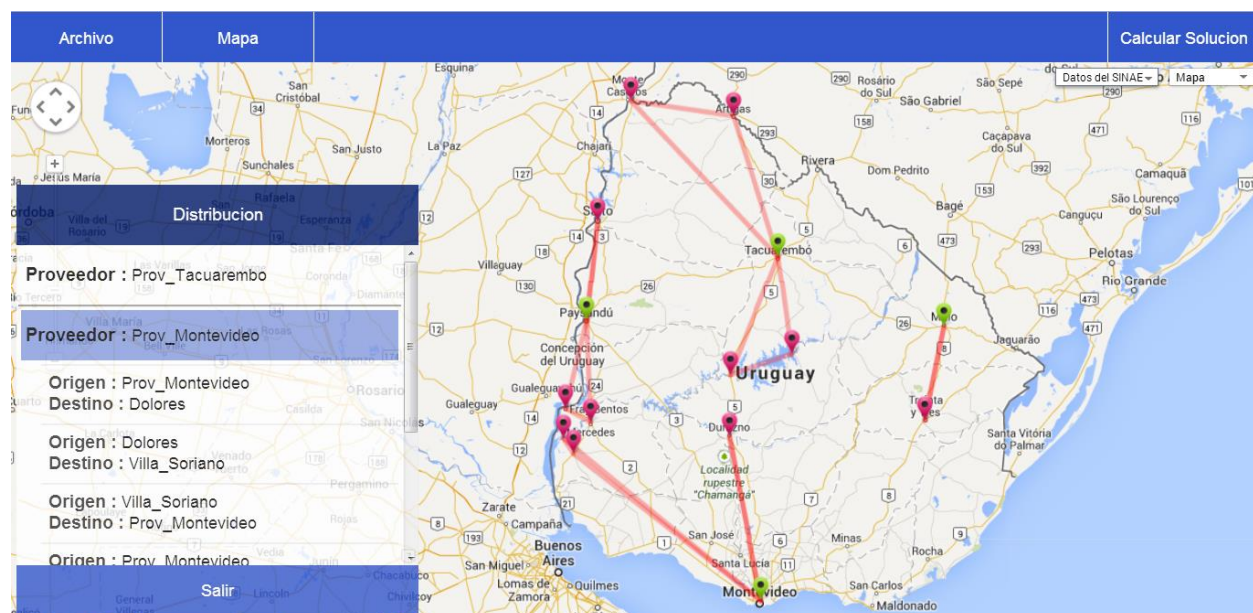
Calcular solución (distribución)

Una vez que se ubican todos los nodos, para calcular la distribución de suministros, presione el botón 'Calcular solución' que se encuentra en la parte superior derecha de la pantalla. Luego de unos segundos, se despliegan en el mapa las líneas que determinan gráficamente la distribución.

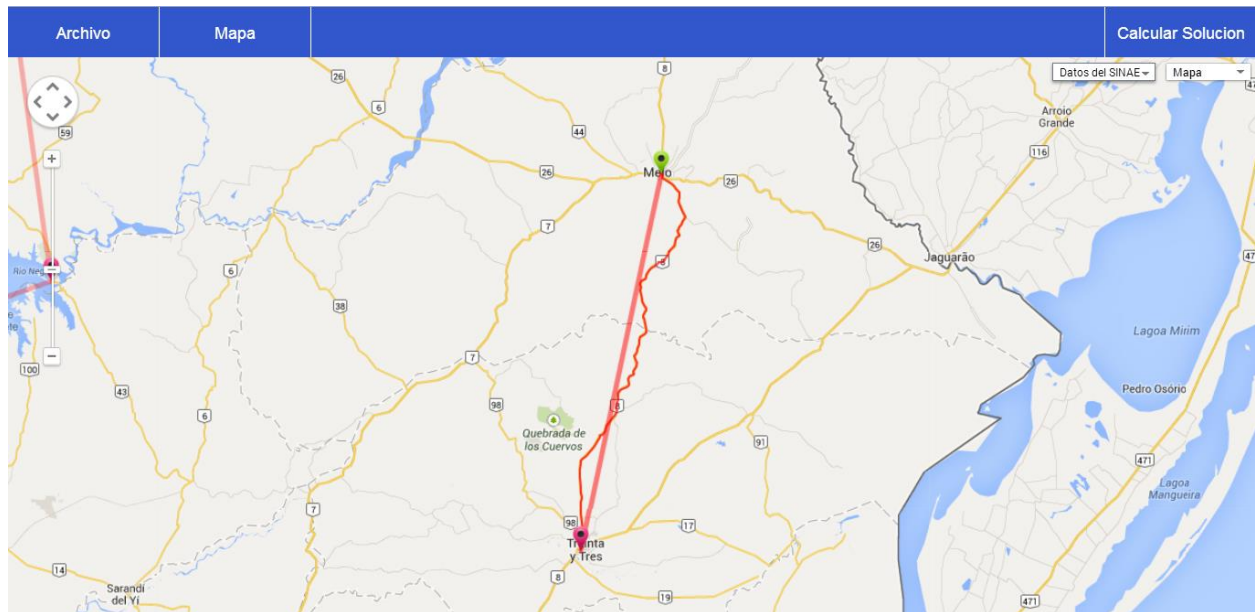


Visualizar distribución (tramos)

Para visualizar el detalle de cada tramo de la distribución se selecciona la opción 'Distribución' en el menú 'Mapa' de la barra superior izquierda de la pantalla. A continuación se despliega una lista con los nombre de los proveedores del mapa



Para ver el detalle, se hace click sobre uno de los proveedores. A continuación se despliega la lista con los demandantes asociados en la distribución, en el orden que deben ser visitados. Al hacer click sobre alguno de estos demandantes, en el mapa principal, se resalta el tramo correspondiente indicando las calles, caminos y/o rutas que deben tomarse para optimizar el tiempo de respuesta. A continuación se muestra una imagen similar



Visualización de datos del SINAÉ (servicio WMS)

Para ver los datos históricos del servicio WMS que ofrece el SINAÉ, se hace click en el botón Datos SINAÉ que se encuentra en la parte superior derecha del mapa. A continuación se despliegan las opciones disponibles, se debe seleccionar el tipo de dato que se quiere visualizar.