

Asignación de Hospitales y Distribución de Ayuda en Logística de Emergencias

Alan Dávila de León, Eduardo Lalla-Ruiz, Christopher Expósito-Izquierdo,
Belén Melián-Batista, and J. Marcos Moreno-Vega

Departamento de Ingeniería Informática y de Sistemas
Universidad de La Laguna
38271 La Laguna, Spain
{alan.davila.de.26, cexposit, elalla, mbmelian, jmmoreno}@ull.es

Resumen Los gobiernos e instituciones internacionales buscan reducir y mitigar el sufrimiento de la población ante desastres naturales o generados por el hombre. El rápido despliegue y desarrollo de una red de suministro de ayuda así como su planificación es crucial para salvar vidas y aliviar los inconvenientes ocasionados a la población. Este tipo de situaciones, por tanto, requiere de una respuesta rápida y adecuada. En este trabajo se presenta y aborda el problema de asignación de hospitales y diseño de rutas de reparto de ayuda buscando minimizar el tiempo de espera de las víctimas existentes. Para su resolución se propone un algoritmo basado en el Recocido Simulado. La experiencia computacional demuestra que se obtienen soluciones de calidad en tiempos de computación reducidos.

Keywords: Logística de emergencias, Reparto de ayuda, Metaheurísticas, Recocido Simulado

1. Introducción

En un reciente estudio del Centre for Research on the Epidemiology of Disasters (CRED¹), en el período comprendido entre los años 1994 y 2013 se registraron 6873 desastres naturales en todo el mundo. Durante este período, cada año se produjeron de media 68000 muertes y 218 millones de personas se vieron afectadas. Además, se produjeron pérdidas económicas valoradas en 147 mil millones de dólares cada año. Por otra parte, debido a conflictos o guerras, el número medio de refugiados al año ha sido de 13 millones y 20 millones de desplazamientos internos. Sin embargo, según un informe de la Federación Internacional de Sociedades de la Cruz Roja (IFRC)² realizado tras una inundación en el 2014 en Afganistán [7], la planificación de la distribución de recursos de socorro fue una de las tareas más difíciles. En dicho estudio se apunta que, a pesar de la complejidad que implica la logística de emergencias, la mayoría de los procesos y

¹ <http://www.cred.be/>

² <http://www.ifrc.org/>

planificaciones se siguen realizando de forma manual, a pesar de que han demostrado ser ineficientes e inadecuados [9]. Por último, se demanda una mejora de los mecanismos de respuesta, específicamente en la planificación, coordinación y entrega de estos recursos, dado que los pocos sistemas existentes no son lo suficientemente flexibles y dinámicos para las situaciones de emergencia donde se desean utilizar.

En las redes de reparto de ayuda gestionadas por instituciones y operadores se busca mitigar los daños ocasionados a la población a través de la distribución de ayuda en forma de medicinas, alimentos, generadores de electricidad, servicios médicos, etc. La cadena de suministro se planifica teniendo en cuenta que se tienen recursos limitados y que cada tipo de demanda tiene un grado de urgencia y ventana de tiempo. Por otra parte, los escenarios de emergencia son altamente dinámicos, lo que obliga a los operadores encargados de gestionar el reparto de ayuda a tomar decisiones rápidas bajo mucha presión. En este contexto, contar con herramientas para ayudar a los gestores en la toma de decisiones y en el diseño de una red de distribución de ayuda es crucial para salvar vidas y aliviar el sufrimiento [1,2,6,8].

En este trabajo se estudia el Problema de Asignación de Hospitales y Distribución de Ayuda (PAHDA) en logística de emergencias. Dado el contexto de este problema, en el que el tiempo es un factor crucial para la resolución del mismo, se propone un algoritmo aproximado basado en el Recocido Simulado. Para medir la efectividad del método se propone un conjunto de instancias.

El resto de este trabajo está organizado de la siguiente forma. La Sección 2 introduce el PAHDA. En la Sección 3 se presenta el recocido simulado propuesto para la resolución de este problema. La experiencia computacional realizada en este trabajo se describe en la Sección 4. Finalmente, en la Sección 5 se enumeran las principales conclusiones y líneas de trabajo futuro.

2. Descripción del Problema

En el Problema de Asignación de Hospitales y Distribución de Ayuda (PAHDA) se tiene un conjunto de víctimas $N = \{1, 2, \dots, n\}$. Para cada víctima $c \in N$ se proporciona el tiempo necesario para ser asistida $d_c > 0$ y la cantidad de ayuda que solicita $q_c > 0$. Por otra parte, se dispone de un conjunto de posibles localizaciones $L = \{1, 2, \dots, m\}$ donde poder instalar h hospitales. Cada localización $l \in L$ tiene asociado el tiempo requerido para instalar un hospital, $t_l > 0$. Además, cada localización $l \in L$ escogida como hospital tiene un conjunto homogéneo de vehículos $V_l = \{1, 2, \dots, v_l\}$ donde cada vehículo $v \in V_l$, dispone de una capacidad $Q > 0$ para transportar la carga. En el PAHDA se busca determinar un subconjunto de h localizaciones donde ubicar hospitales y desde los cuales iniciar las rutas de reparto de ayuda humanitaria usando para ello los vehículos de cada hospital.

Este problema cuenta con una serie de restricciones que deben cumplirse para que una posible solución sea considerada factible:

- (1) Todas las víctimas deben ser atendidas

- (II) Cada vehículo está asociado a un hospital, que es el origen y punto final de su ruta de reparto de ayuda
- (III) Cada víctima debe ser atendida una única vez
- (IV) La carga es indivisible. Esto es, una víctima solo puede ser visitada por un vehículo y, por tanto, no puede recibir ayuda de forma parcial
- (V) La demanda de las víctimas asociadas a la ruta de cada vehículo no puede sobrepasar la capacidad del mismo

El PAHDA se puede representar como un grafo completo $G = (H, A)$ con $n + m$ vértices. Donde H es el conjunto de vértices compuesto por las víctimas y posibles localizaciones, $H = N \cup L$. Por otra parte, A representa el conjunto de arcos entre cada par de vértices $(i, j) \in H$. Estos arcos representan el mejor camino para ir desde el nodo i hasta el nodo j y tiene asociado un costo con un determinado costo a_{ij} y tiempo de viaje t_{ij} .

Para determinar la calidad de las soluciones, se utiliza el tiempo total de espera de las víctimas hasta que reciben la ayuda. Es decir, dada una solución s , su valor de función objetivo vendría establecido por la siguiente expresión:

$$f(s) = \sum_{l \in L'} \sum_{r \in R_l} \sum_{i=0}^{n_r} w(\sigma_i^r), \tag{1}$$

donde L' denota el subconjunto de localizaciones seleccionadas para instalar el hospital, R_l denota el conjunto de rutas salientes desde cada hospital $l \in L'$, n_r es el número de víctimas visitas en cada ruta r y $w(\sigma_i^r)$ es el tiempo de espera acumulado para la víctima σ_i^r :

$$w(\sigma_i^r) = \begin{cases} t_{\sigma_i^r} & \text{si } i = 0 \\ w(\sigma_{i-1}^r) + t_{\sigma_{i-1}^r, \sigma_i^r} + d_{\sigma_i^r} & \text{en otro caso} \end{cases} \tag{2}$$

En este problema se persigue minimizar la función objetivo (2). Esto es, minimizar la suma de los tiempos de espera de las víctimas. Es por ello, que se propone minimizar el tiempo de espera de la última víctima atendida. Nótese que, por tanto, no se tiene en cuenta el último desplazamiento que deben realizar los vehículos para regresar a sus respectivas localizaciones de origen.

Con el objetivo de facilitar el entendimiento de este problema, en la Figura 1 se proporciona un ejemplo gráfico de un posible escenario (izquierda) y una solución al mismo (derecha). El problema inicial consta de 7 posibles localizaciones y 17 víctimas. En este escenario se persigue levantar 2 hospitales y donde cada hospital cuenta con 2 vehículos. La solución generada se compone por cuatro rutas de reparto de ayuda que parten de las dos localizaciones seleccionadas. Cabe destacarse que el usuario proporciona el número de localizaciones a seleccionar, en este caso, $h = 2$.

3. Propuesta Algorítmica

En los momentos posteriores a un desastre, los encargados de hacer frente al mismo tienen que determinar rápidamente una red de distribución de ayuda

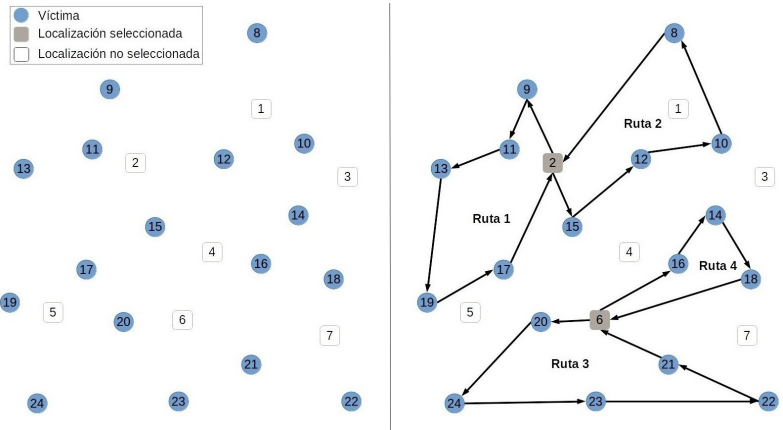


Figura 1: Ejemplo de un escenario (izquierda) y de una solución (derecha) para el problema de asignación de hospitales y distribución de ayuda

eficiente. Incluso, aunque inicialmente se dispusieran de muchos recursos, los gestores buscarán dividir la zona del desastre en sectores así como limitar el número de agentes en cada sector con el objetivo de llegar a todas las víctimas en el menor tiempo posible. Por tanto, es necesario disponer de algoritmos rápidos y que provean soluciones de alta calidad. Con este fin, en esta sección se propone una metaheurística basada en Recocido Simulado.

3.1. Recocido Simulado

El Recocido Simulado (SA, [3]) es una metaheurística que, a diferencia de la Búsqueda Local, se permiten movimientos que empeoren el valor de función objetivo, con el fin de evitar el estancamiento en óptimo locales. Estos movimientos se realizan de manera controlada cuando la búsqueda avanza hacia soluciones de alta calidad. Para decidir cuándo se acepta un movimiento que empeora el valor de función objetivo se utiliza una función de probabilidad $P(s, s')$ donde s es una solución del problema y s' una solución vecina. Como esta función depende de la temperatura, t , la probabilidad de aceptar un mal movimiento a medida que avanza la búsqueda disminuye:

$$P(s, s') = e^{(f(s)-f(s'))/t} \tag{3}$$

En Algoritmo 1 se ilustra el ya conocido algoritmo del Recocido Simulado, donde como se puede observar la aceptación de la degradación sigue la fórmula $rand(0, 1) < P(s_{actual}, s_{vecina})$.

La solución de comienzo se genera mediante un procedimiento de búsqueda adaptivo, voraz y aleatorio (GRASP, del inglés Greedy Randomized Adaptive Search Procedure [4]), donde la lista se genera en base a la media de las distancias desde cada localización (no incluida en la solución) hasta las víctimas.

Algoritmo 1 Recocido Simulado, SA

```

t ← temperatura inicial de acuerdo a (4)
sactual = generarSoluciónInicial()
while t > tf do
  while iter < iterMax do
    svecina = s' ∈ N(sactual)
    if f(svecina) < f(sactual) then
      sactual = svecina
    else
      if rand(0,1) < P(sactual, svecina) then
        sactual = svecina
      end if
    end if
    iter = iter + 1
  end while
  actualizarTemperatura(t)
end while
devolver sactual

```

De esta lista, se obtiene la lista restringida (LRC) de tamaño h y se escoge una localización al azar, hasta completar las h localizaciones necesarias.

Para la generación de soluciones vecinas se utiliza el movimiento de reinsertión. Esto es, se extrae una víctima dentro de una ruta de distribución de ayuda y se inserta en otra distinta.

Para conseguir una buena diversificación al inicio de la búsqueda, la temperatura inicial, t_0 , se calcula de forma interna por la metaheurística. Para ello, se generan n soluciones vecinas y se selecciona aquella con el mejor valor encontrado, s , y aquella con el peor valor de la función objetivo encontrado, s' , luego se calcula la temperatura inicial de acuerdo a la siguiente fórmula:

$$t_0 = (f(s') - f(s)) \cdot 0,95 \quad (4)$$

Por otra parte, el número de iteraciones máximas, $iterMax$, que realiza la metaheurística corresponde al número de víctimas presentes en el problema, con lo que se pretende adaptar la búsqueda a la dimensión de la instancia a resolver, *i.e.*, $iter = n$.

3.2. Recocido Simulado con Recalentamiento

En este trabajo, se propone una versión adicional del Recocido Simulado con Recalentamientos (SA-R) para resolver el PAHDA. Las modificaciones sobre la versión propuesta en la subsección anterior son las presentadas a continuación:

- Cada vez que se acepta una solución vecina se disminuye la temperatura. Con esto se intenta evitar movimientos que empeoren el valor objetivo a medida que la búsqueda se acerca a regiones del espacio de búsqueda con soluciones de alta calidad.

- El proceso de recalentamiento tiene por objetivo, una vez la temperatura ha disminuido tanto que ya únicamente se permiten movimientos de mejora, comenzar de nuevo la búsqueda partiendo desde una solución vecina a la mejor solución encontrada hasta el momento.

En el Algoritmo 2 se muestra el pseudo-código de SA-R. Tal como se puede comprobar, se incluye un recalentamiento que implica la generación de una solución de comienzo mediante el procedimiento de *agitación*(s_{mejor}). Este procedimiento consiste en generar una solución vecina de forma aleatoria de la mejor solución conocida s_{mejor} a lo largo de la búsqueda.

Algoritmo 2 Recocido Simulado con Recalentamiento, SA-R

```

recalentar = 0
t ← temperatura inicial de acuerdo a (4)
sactual = generarSoluciónInicial()
while recalentar < maxRecalentar do
  t = calcularTemperatura()
  while t > tf do
    iter = 0
    while iter < iterMax do
      svecina = s' ∈ N(sactual)
      if rand(0, 1) < P(sactual, svecina) then
        sactual = svecina
        actualizarTemperatura(t)
        if f(sactual) < f(smejor) then
          smejor = sactual
        end if
      end if
      iter = iter + 1
    end while
  end while
  recalentar = recalentar + 1
  sactual = agitación(smejor)
end while
devolver smejor

```

4. Resultados Computacionales

En esta sección se presenta el estudio computacional realizado para evaluar el rendimiento de los procedimientos para resolver el PAHDA. Las características del equipo donde se han realizado las experimentaciones son las siguientes:

- Memoria RAM: 8 GB
- Memoria caché: 6 MB
- Procesador: Intel Core i7-3632QM CPU @ 2.20 GHz x 8

- SO: Linux, Ubuntu 14.04 LTS 64 bits
- Número de ejecuciones por instancia: 10

Para esta experiencia computacional se generó un conjunto de escenarios con distintas características, tales como el número de víctimas, el número de posibles localizaciones donde situar los puntos de reparto de ayuda, número de hospitales y vehículos:

- Número de instancias: 18 instancias
 - Número de localizaciones: 10, 20, 50
 - Número de víctimas: 25, 100, 500
- Número de hospitales: 4 hospitales
- Número de vehículos: 3 vehículos por hospital

4.1. Selección de Parámetros

Con objeto de analizar la contribución de Recocido Simulado con Recalentamientos (SA-R) se implementaron los siguientes métodos:

1. Búsqueda por Entornos Variables (VNS, del inglés [5]). Utiliza 2 estructuras de entorno, *i.e.*, intercambio y re inserción, que varían según el valor de $k = \{1, 2\}$, respectivamente. La solución de partida es generada por el algoritmo constructivo GRASP propuesto en la Sección 3.1. El procedimiento de la búsqueda se repite un número máximo de iteraciones *iterMax*.
2. Recocido simulado: se implementa una versión básica de esta metaheurística como se describe en la Sección 3.1.

Previo a la comparativa general para analizar la contribución de SA-R respecto a los métodos anteriormente mencionados, en esta subsección se realiza un estudio previo sobre los parámetros para el SA y la VNS.

Cuadro 1: Ajuste del valor de la temperatura final (t_f) del SA en base a la temperatura inicial t_0 o a valores prefijados

t_f	Min	Media	Max	t (ms.)
$t_0 \cdot 10^{-9}$	917936.19	926260.16	934544.13	12541.56
$t_0 \cdot 10^{-7}$	918552.22	926260.71	936055.60	9788.78
$t_0 \cdot 10^{-5}$	919731.25	926731.13	937515.52	7452.94
0,01	918095.60	925395.09	933447.86	13476.11
0,10	917892.70	926150.43	933632.97	12070.17
1,00	918498.23	926863.66	935058.63	10777.5
10,0	918683.14	926325.30	934903.49	9464.67
100	919444.56	926812.41	935592.52	8290.56

En la comparativa mostrada en el Cuadro 1 se muestran los valores medios para el conjunto de instancias considerado en este trabajo. Se reportan los

valores mínimos, *Min*, medios, *Media* y los valores máximos *Max*, así como el tiempo computacional, $t(ms)$ medido en milisegundos. Como se puede observar no parece determinante fijar un valor predeterminado de parada o dependiente de la temperatura inicial, Por lo que se ha optado por fijar t_f a 0.01 dado que presenta los valores más competitivos en términos de valor medio y máximo. Por último, el número de recalentamiento máximos, *maxRecalentar* de SA-R se fija a 3 para mantener un equilibrio entre tiempo de ejecución y calidad de las soluciones proporcionadas por el método.

Cuadro 2: Análisis de la calidad de las soluciones proporcionadas por la VNS dependiendo del número de iteraciones, *iterMax*

<i>iterMax</i>	Min	Media	Max	t (ms.)
10	929939.86	938935.75	949520.57	13771.11
50	928513.99	935498.02	944479.28	34943.83
100	925392.96	934249.33	943834.59	57651.33

En el Cuadro 2 se muestran los resultados obtenidos al incrementar el número de iteraciones de la VNS. Como se puede apreciar, a medida que crece el número de iteraciones también decrece el valor objetivo de las soluciones, pero con un aumento en el tiempo de cómputo. Buscando un equilibrio entre la calidad de las soluciones y el tiempo de cómputo necesario para proporcionarlas, y así realizar una justa comparación con SA, se decide fijar *iterMax* a 50 iteraciones.

4.2. Comparativa entre Métodos de Resolución

En el Cuadro 3 se muestra la comparativa entre los métodos implementados en este trabajo una vez se han escogido los parámetros de los mismos (ver Sección 4.1). Cada fila representa una instancia, excepto en el caso de la última fila, la cual representa los valores promedios correspondientes. En la columna *Instancia* se muestran las características principales de cada instancia, esto es, número de localizaciones m , número de víctimas n , y número de hospitales h . Por otra parte, para cada método (*i.e.*, VNS, SA o SA-R) se reportan los valores mínimos, *Min*, valores medios, *Media*, y los valores máximos, *Max*. Además se reporta el tiempo de cómputo en milisegundos en la columna, $t (ms.)$.

Como se observa en el Cuadro 3, SA-R proporciona soluciones de mejor calidad que SA y VNS. Además, se puede observar que SA-R es capaz de proporcionar un mayor número de mejores soluciones (marcadas en negrita) para las instancias consideradas. Esto implica que ante distintos escenarios su utilización es más apropiada que las otras propuestas. Por último, tanto SA como SA-R presentan un comportamiento competitivo tanto en tiempo de computación como en calidad respecto a la VNS. Esto aconseja el uso del Recocido Simulado para este tipo de problemas, en especial, la propuesta que implementa recalentamientos.

Cuadro 3: Comparativa entre métodos de resolución para el PAHDA. Mejor solución en negrita

Instancia	VNS				SA				SA-R						
	m	n	h		Min	Media	Máx.	t (ms.)	Min	Media	Max	t (ms.)	Min	Media	Max
10	25	4	20436	21269	23617	19	20495	22361	24597	87	20480	22295	22837	241	
10	25	4	18881	19409	20909	19	18227	19387	20331	82	17296	19040	20888	245	
20	100	4	158621	163380	170559	593	157778	161119	166692	1099	157697	160965	164265	3178	
20	100	4	147036	151371	158237	539	145367	149979	156273	1105	145387	151193	159860	3227	
50	500	4	2736514	2776191	2815582	102040	2715007	2733565	2762412	30037	2700425	2733493	2770820	88601	
50	500	4	2612291	2642181	2665763	98478	2592894	2623607	2678511	28852	2583646	2612336	2657553	88243	
10	25	4	17240	17821	19066	24	17235	18062	20420	88	17181	17778	18869	298	
10	25	4	18144	19139	20328	20	18238	19518	21296	87	18187	20014	21330	283	
20	100	4	142086	145285	152024	692	142144	144772	147623	1143	141856	144652	147003	3366	
20	100	4	139409	146222	150200	638	141647	146437	150153	1033	141058	144886	148516	3002	
50	500	4	2671241	2692347	2708271	110450	2662034	2672471	2686411	30599	2652448	2661230	2670095	91296	
50	500	4	2499055	2513331	2527547	106548	2477562	2488377	2501183	30808	2454305	2474451	2488398	87884	
10	25	4	17683	18409	19281	39	17611	18239	19184	98	17595	18061	18367	310	
10	25	4	14609	15620	16882	34	14704	15661	17065	98	14601	15648	17760	312	
20	100	4	126763	130733	136221	945	127402	130159	133676	1083	128549	129758	131703	3402	
20	100	4	143828	145185	147206	903	142675	143529	144530	1186	142208	143244	144135	3470	
50	500	4	2502149	2516945	2535424	128363	2471852	2483308	2500078	31512	2474001	2487841	2506685	94923	
50	500	4	2700681	2714730	2731014	118485	2674930	2684246	2698148	30213	2669746	2676557	2690997	88864	
				927037.05	936087.11	945451.72	37157.16	919877.88	926377.61	936032.38	10511.66	916481.44	924080.11	933337.83	31174.72

5. Conclusiones y Líneas de Trabajo Futuras

En este trabajo, se considera el Problema de Asignación de Hospitales y Distribución de Ayuda dentro de la logística de emergencias. En este problema se busca asignar hospitales y determinar las rutas de distribución de ayuda en zonas de desastre. Para su resolución se propone un algoritmo basado en el Recocido Simulado con Recalentamiento (SA-R). El rendimiento de SA-R se analiza en comparación con otros métodos aproximados tales como la versión básica del SA o la búsqueda por entornos variables. De este análisis, se puede concluir que el rendimiento de SA-R en términos de la calidad de las soluciones proporcionadas es superior a los otros dos métodos. Por otra parte, SA-R es capaz de proveer soluciones de alta calidad en menos de dos minutos de media, lo que lo hace apropiado y competitivo a la hora de atajar este tipo de problemas, donde el tiempo juega un papel crucial.

Como trabajo futuro, se plantea el desarrollo de un modelo matemático así como el diseño de un sistema de ayuda a la decisión que permita visualizar, analizar y modificar las soluciones previamente proporcionadas por distintos algoritmos. Por otra parte, también se propone como trabajo futuro la generación de una batería de instancias utilizando los datos abiertos y disponibles de desastres reales (*e.g.*, Katrina, Fukushima, etc.).

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Economía y Competitividad de España (project TIN2012-32608). Eduardo Lalla-Ruiz y Christopher Expósito-Izquierdo agradecen al Gobierno de Canarias el apoyo financiero que reciben a través de sus becas de doctorado.

Referencias

1. Nezhil Altay and Walter G Green. Or/ms research in disaster operations management. *European journal of operational research*, 175(1):475–493, 2006.
2. B. Balcik, B. and Beamon. Facility location in humanitarian relief. *International Journal of Logistics*, 11(2):101–121, 2008.
3. Lawrence Davis. Genetic algorithms and simulated annealing. 1987.
4. T.A. Feo and M.G.C. Resende. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6(2):109–133, 1995.
5. P. Hansen, N. Mladenović, and J. Pérez. Variable neighbourhood search: methods and applications. *Annals of Operations Research*, 175(1):367–407, 2010.
6. E. Luis, I. S. Dolinskaya, and K. R. Smilowitz. Disaster relief routing: Integrating research and practice. *Socio-economic planning sciences*, 46(1):88–97, 2012.
7. Stephane M. After action review of dref operation # mdraf002 - afghanistan floods and landslide. Technical report, International Federation of Red Cross and Afghan Red Crescent Society, 2014.
8. N. Simpson and P. Hancock. Fifty years of operational research and emergency response. *Journal of the Operational Research Society*, pages S126–S139, 2009.
9. A. Thomas and Logistics Humanitarian. Enabling disaster response. *Institute Fritz*, 2007.