

## Ciencias de la Educación y Humanidades

### EL MÉTODO DE PATH RELINKING EN LA BÚSQUEDA DE LA RUTA MÁS CORTA

### THE PATH RELINKING METHOD IN THE SHORTEST PATH SEARCH.

Rudys de Jesús Martínez

#### RESUMEN

En este artículo se presenta el método de Path Relinking o de re-encadenamiento de trayectorias, el cual sirve para calcular la ruta más corta mediante un algoritmo que genera iteraciones a gran velocidad y de mucha eficacia encontrando rutas cada vez mejores. Es utilizado con el fin de optimizar los costos a través de la búsqueda de nuevas trayectorias. Este algoritmo es muy eficiente en su búsqueda con el fin de mejorarla y se pueden calcular rutas que manualmente serán casi imposibles de resolver y que con él se puede lograr. Este método se utiliza en áreas como Investigación de Operaciones, las ingenierías y Ciencias de la Computación. Es utilizado eficientemente para calcular rutas más factibles como en transporte, horarios de operadores telefónicos, plan de rutas de la ciudad, redes eléctricas, diseño de las rutas para los vehículos, planes de producción, entre otros.

**PALABRAS CLAVE: RUTA MÁS CORTA, ALGORITMO, RE-ENCADENAMIENTO, TRAYECTORIA, OPTIMIZACIÓN, ENCADENAMIENTO DE TRAYECTORIAS.**

#### ABSTRACT

In this paper the method of Path relinking or relinking paths, which is used to calculate the shortest route using an algorithm that generates iterations at high speed and very effectively finding it increasingly appears trails. It is used in order to optimize costs through the search for new paths. This algorithm is very efficient in its search to improve and you can manually calculate routes that are almost impossible to resolve and that it can be achieved. This method is used in areas such as operations research, engineering and computer science. It is more efficiently used to calculate feasible routes and transport schedules telephone operators, plan routes in the city, power grids, design routes for vehicles, production plans, among others.

**KEYWORDS: SHORTEST PATH, ALGORITHM, RE-CHAINING, TRAJECTORY, OPTIMIZATION, PATH CHAINING.**

#### INTRODUCCIÓN

En este artículo se presenta el método de Path Relinking, el cual es de mucha importancia porque es un programa con la destreza de calcular la ruta más corta contemplando así la posibilidad de gastar menos en

1. Docente, Departamento de Educación y Humanidades, Facultad Regional Multidisciplinaria Matagalpa, UNAN-Managua. ORCID: 0009-0006-5566-1123 .  
Correo electrónico: [rjmartinez04@yahoo.com](mailto:rjmartinez04@yahoo.com)

## Ciencias de la Educación y Humanidades

distintas situaciones. Puede resolver de manera rápida cualquier problema, ya que se pueden formular como modelos de redes logrando soluciones enteras sin restringirlas, pero también pueden darse no enteras. Se trata de ir buscando mediante las diferentes iteraciones que realiza él mismo el recorrido más corto, por ende, más factible para una empresa particularmente. La situación o problema puede ser muy grande, pero el programa lo resuelve en pequeños algoritmos y en un tiempo corto.

El problema de la ruta más corta es de mucha utilidad en áreas como Investigación de Operaciones, Ciencias de la Computación y las diferentes Ingenierías. Su gran variedad en aplicaciones como el envío de materiales entre dos puntos concretos por ejemplo la distribución del correo, lo recorridos que realizan las rutas urbanas de una ciudad.

Algunas aplicaciones de este método en situaciones prácticas son: transporte, horarios de operadores telefónicos, plan de rutas de la ciudad, redes eléctricas, diseño de las rutas para los vehículos, planes de producción, entre otros.

Los resultados que se obtienen al utilizar este método de optimización combinatoria se basan en el concepto de encadenamiento de trayectorias, es un proceso de refinamiento y mejora de las soluciones encontradas, tiene un excelente desempeño en su capacidad para encontrar soluciones de alta calidad, en sistemas matemáticamente complejos, y de tamaños muy grande.

El método de Path Relinking se caracteriza principalmente por descubrir nuevas búsquedas en subespacios de soluciones predefinidos, que no se han explorado mucho, pero que son muy atractivos por estar cerca de óptimas soluciones conocidas. Para llevarlo a cabo se ubica en la región de interés que no se ha explorado tanto, pero que es de buena calidad, la cual va recibiendo atributos de otras soluciones ya conocidas, también de calidad, a la que se le nombra solución guía.

Este método de la ruta más corta es un método de programación lineal que trata de encontrar la solución a problemas de optimización, que resulta de una combinatoria y diferentes aplicaciones. Su objetivo principal estriba en encontrar las rutas más cortas o de costos mínimos, según sea el caso que inicia desde un nodo en particular hasta otro de la red.

El Path Relinking se inicia con el propósito de construir una solución de prueba a través de la aplicación de un método rápido, eficiente, aleatorio, y que después permita aplicar una búsqueda local a la solución encontrada. Este procedimiento será repetido tantas veces como sea necesario con el fin de alcanzar un criterio de "parada" que será el número máximo de iteraciones, tiempo de ejecución, caminos cortos o máximos, entre otros. El más óptimo local encontrado durante todo este proceso será el que se regresa como la mejor solución encontrada por el método.

El método se representa mediante una red compuesta por nodos y líneas que les unen. Un nodo se representa gráficamente en forma de círculo, cada uno de ellos son de mucha importancia, ya que denotan los orígenes y destinos del problema a desarrollar. De igual forma la red representa un conjunto de líneas que conectan pares de puntos, los puntos reciben el nombre de nodos o vértices y las líneas aristas o arcos, los cuales pueden tener una dirección asociada, en este caso se llaman arcos dirigidos. A continuación, una figura 1 con esta representación.

## Ciencias de la Educación y Humanidades

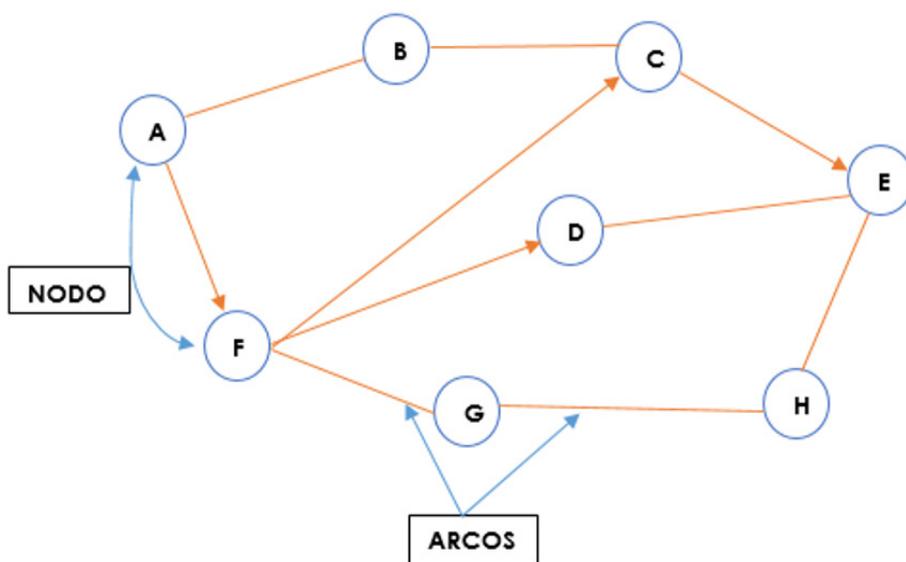


Figura 1. Representación de redes con nodos y aristas

Para Santamaría (2004), el método de Path Relinking (PR) fue propuesto originalmente dentro del contexto de Búsqueda Tabú (BT) [1,2], en el cual uno de los principales objetivos es efectuar un equilibrio entre la búsqueda en intensificación y diversificación. Existen algunas características que se han agregado a la Búsqueda Dispersa (BD), como extensión de su filosofía básica, y son al mismo tiempo incluidas en el contexto de Path Relinking.

### MATERIALES Y MÉTODOS

Para Santamaría (2004), el método de Path Relinking (PR) fue propuesto originalmente dentro del contexto de Búsqueda Tabú (BT) [1,2], en el cual uno de los principales objetivos es efectuar un equilibrio entre la búsqueda en intensificación y diversificación. Existen algunas características que se han agregado a la Búsqueda Dispersa (BD), como extensión de su filosofía básica, y son al mismo tiempo incluidas en el contexto de Path Relinking.

## Ciencias de la Educación y Humanidades

### RESULTADOS

El Path Relinking o re-encadenamiento de trayectorias está fundamentado en generar nuevas soluciones a través de la exploración de nuevas trayectorias que las descubren y que sean de buena calidad, comenzando esta búsqueda a partir de una de estas soluciones, a la cual se le llama solución inicial. A partir de ésta se va generando un camino en el espacio del vecindario que dirige la búsqueda de diferentes soluciones, designadas como soluciones guías.

Este re-encadenamiento de trayectorias se basa en que entre dos soluciones se puede trazar un camino que les una, de tal forma que las soluciones de dicho camino tengan atributos de las primeras. Luego para continuar generando nuevos caminos se hace necesaria la selección de movimientos que cumplan con algunos objetivos como introducir de forma progresiva atributos iniciando con una solución inicial y continuando con la solución guía, tratando de disminuir la distancia entre las condiciones de estas soluciones. Los papeles de ambas soluciones se pueden intercambiar, y también se pueden mover hacia las otras como una forma de generar nuevas combinaciones.

En la figura siguiente se puede observar un ejemplo del método Path Relinking en el cual a partir de un camino inicial entre dos soluciones que están unidas por una línea continua se construye un nuevo camino para volver a unir las dos soluciones. El propósito es poder encontrar nuevas soluciones que a simple vista son casi imposibles, es como buscar a ciegas, y que al agregar atributos de soluciones conocidas se pueda llegar a mejorarla cada vez. Continuando con el ejemplo la línea punteada (figura 2) es un camino cualquiera generado para unir las dos soluciones. El nodo de color es una mejor solución que las anteriores.

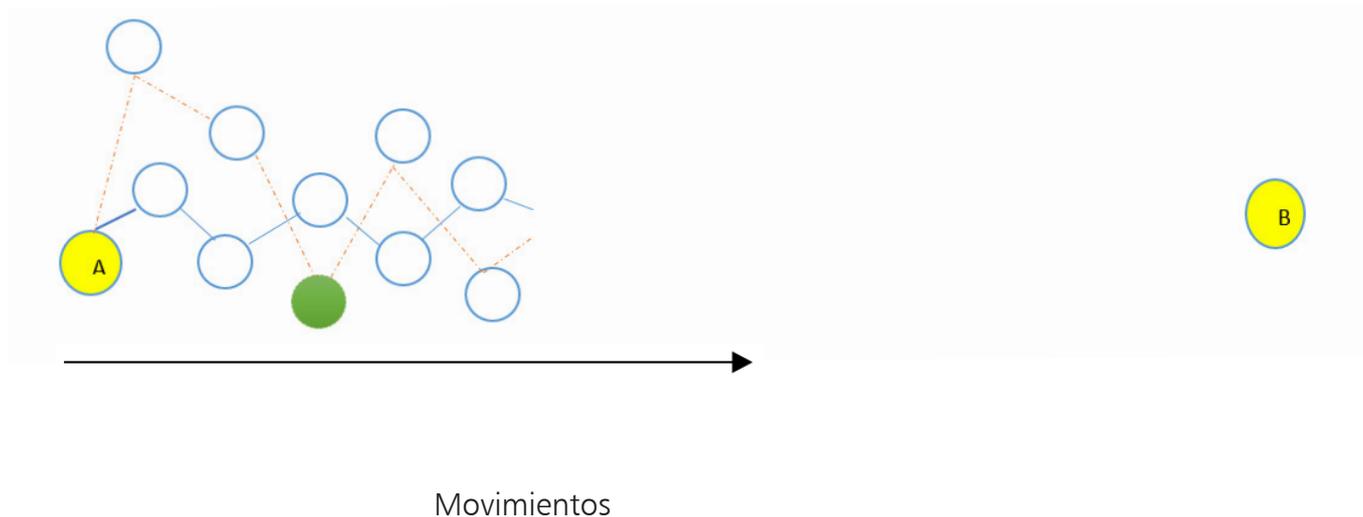


Figura 2. Representación del método Path Relinking

El programa como veremos en algunas imágenes hace una búsqueda a gran velocidad aleatoria de la mejor forma posible, cuando se le da la orden de realizar otra búsqueda lo hace repetitivamente cuantas veces sea necesario con una certeza impresionante. Hay un instante donde ya no hay ningún posible

## Ciencias de la Educación y Humanidades

camino a seguir, el mismo programa se detiene porque ya no hay ninguna ruta que buscar.

A continuación, se presenta un ejemplo de la ruta más corta, con la matriz de diecisiete ciudades (gr17), para viajar de una ciudad a otra, y en él se representarán las posibles iteraciones que el programa efectúa. (figura3)

Sea la matriz:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
A	0	633	257	91	412	150	80	134	259	505	353	324	70	211	268	246	121
B	633	0	390	661	227	488	572	530	555	289	282	638	567	466	420	745	518
C	257	390	0	228	169	112	196	154	372	262	110	437	191	74	53	472	142
D	91	661	228	0	383	120	77	105	175	476	324	240	27	182	239	237	84
E	412	227	169	383	0	267	351	309	338	196	61	421	346	243	199	528	297
F	150	488	112	120	267	0	63	34	264	360	208	329	83	105	123	364	35
G	80	572	196	77	351	63	0	29	232	444	292	297	47	150	207	332	29
H	134	530	154	105	309	34	29	0	249	402	250	314	68	108	165	349	36
I	259	555	372	175	338	264	232	249	0	495	352	95	189	326	383	202	236
J	505	289	262	476	196	360	444	402	495	0	154	578	439	336	240	685	390
K	353	282	110	324	61	208	292	250	352	154	0	435	287	184	140	542	238
L	324	638	437	240	421	329	297	314	95	578	435	0	254	391	448	157	301
M	70	567	191	27	346	83	47	68	189	439	287	254	0	145	202	289	55
N	211	466	74	182	243	105	150	108	326	336	184	391	145	0	57	426	96
O	268	420	53	239	199	123	207	165	383	240	140	448	202	57	0	483	153
P	246	745	472	237	528	364	332	349	202	685	542	157	289	426	483	0	336
Q	121	518	142	84	297	35	29	36	236	390	238	301	55	96	153	336	0

Figura 3. Matriz de la distancia de diecisiete ciudades.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
A	0	633	257	91	412	150	80	134	259	505	353	324	70	211	268	246	121
B	633	0	390	661	227	488	572	530	555	289	282	638	567	466	420	745	518
C	257	390	0	228	169	112	196	154	372	262	110	437	191	74	53	472	142
D	91	661	228	0	383	120	77	105	175	476	324	240	27	182	239	237	84
E	412	227	169	383	0	267	351	309	338	196	61	421	346	243	199	528	297
F	150	488	112	120	267	0	63	34	264	360	208	329	83	105	123	364	35
G	80	572	196	77	351	63	0	29	232	444	292	297	47	150	207	332	29
H	134	530	154	105	309	34	29	0	249	402	250	314	68	108	165	349	36
I	259	555	372	175	338	264	232	249	0	495	352	95	189	326	383	202	236
J	505	289	262	476	196	360	444	402	495	0	154	578	439	336	240	685	390
K	353	282	110	324	61	208	292	250	352	154	0	435	287	184	140	542	238
L	324	638	437	240	421	329	297	314	95	578	435	0	254	391	448	157	301
M	70	567	191	27	346	83	47	68	189	439	287	254	0	145	202	289	55
N	211	466	74	182	243	105	150	108	326	336	184	391	145	0	57	426	96
O	268	420	53	239	199	123	207	165	383	240	140	448	202	57	0	483	153
P	246	745	472	237	528	364	332	349	202	685	542	157	289	426	483	0	336
Q	121	518	142	84	297	35	29	36	236	390	238	301	55	96	153	336	0

Seleccione un origen (A-Q): C  
 Seleccione un destino (A-Q): P  
 La ruta más corta encontrada es C-D-P  
 La distancia a recorrer es de 465  
 Puede que hayan más rutas igual o más cortas que ésta.  
 ¿Desea buscarlas? (s/n): s  
 La ruta más corta encontrada es C-M-D-P  
 La distancia a recorrer es de 455  
 Puede que hayan más rutas igual o más cortas que ésta.  
 ¿Desea buscarlas? (s/n): s  
 ¡Finalizó la búsqueda de la ruta más corta!

Figura 4. Representación de la iteración realizada con el programa Path Relinking

## Ciencias de la Educación y Humanidades

Ejemplo de cálculo del recorrido de la ciudad C a la ciudad P utilizando el método Path Relinking.

En este ejemplo se pide al programa que encuentre la ruta más corta de la ciudad C a la ciudad P. El programa pide un punto de origen o salida y se escribe el nombre de la ciudad, que en este caso es la ciudad C, luego pide el lugar de destino que será la ciudad P. Encuentra la primera ruta y la define como C-D-P dando como resultado 465 de distancia, pasando por tres ciudades. Este resultado se deriva del siguiente recorrido:

Inicia la ruta de la ciudad C a la ciudad D con una distancia de 228, luego va de la ciudad D a la ciudad P donde hay una distancia de 237 para hacer un total de 465. Luego él mismo propone realizar otra búsqueda porque puede haber otra igual o menor que la anterior, considerando que el programa tratará de encontrar una ruta mejor que la anterior, siempre espera unos segundos y encuentra otra ruta que es C-M-D-P con una distancia menor que la anterior de 455. Detalladamente es lo siguiente: la distancia de la ciudad C a la ciudad M es de 191, luego de la ciudad M a la D tiene 27, de la ciudad D a la ciudad P es 237 para un recorrido total de 455, habiendo mejorado la ruta anterior en diez unidades. Nuevamente hace la sugerencia de buscar una nueva ruta, pero más tarde indica que finalizó la búsqueda de la ruta más corta.

A continuación, se presenta como ejemplo uno de los algoritmos en pseudocódigos del Path Relinking tal como lo presenta el programa:

### AUTOBLACKLIST.CS.

```
using System.Collections;
using System.Collections.Generic;
using System.Linq;

namespace PathRelinking
{
    internal class AutoBlackList : IEnumerable<string>
    {
        private List<string> _list = new List<string>();

        public void Add(string item)
        {
            _list.Add(item);
            if (item[item.Length - 1] == GlobalVariables.LastCharBlackList)
            {
                var pattern = item.Substring(0, item.Length - 1);
                var matchingIndexes = new List<int>();
                for (int i = _list.Count - 1; i >= 0; i--)
                {
                    if (_list[i].StartsWith(pattern))
                    {
                        matchingIndexes.Add(i);
                    }
                }
            }
        }
    }
}
```

## Ciencias de la Educación y Humanidades

```

        else
        {
            break;
        }
    }
    if (matchingIndexes.Count == GlobalVariables.LettersUsed.Count - item.Length)
    {
        foreach (var index in matchingIndexes)
        {
            _list.RemoveAt(index);
        }
        _list.Add(pattern);
    }
}

private int _listCount;

public void CleanBlackList()
{
    if (_listCount == _list.Count)
    {
        return;
    }
    do
    {
        _listCount = _list.Count;
        ReSortAndReduce();
    } while (_listCount > _list.Count);
}

private void ReSortAndReduce()
{
    Sort();

    #region Reduce

    for (int i = 0; i < _list.Count; i++)
    {
        var item = _list[i];
        var pattern = item.Substring(0, item.Length - 1);
        var charToCheck = GlobalVariables.LettersUsed.Except(pattern + GlobalVariables.CurrentDestinyStr).Last();
        var lastIndex = _list.LastIndexOf(pattern + charToCheck);
        if (lastIndex == -1)
        {
            while (i < _list.Count && _list[i].Length == pattern.Length + 1 && _list[i].StartsWith(pattern))
            {
                i++;
            }
            continue;
        }
    }
}

```

## Ciencias de la Educación y Humanidades

```
        if (lastIndex < i)
        {
            Sort();
            i = -1;
            continue;
        }
        i = lastIndex;
        item = _list[i];
        var matchingIndexes = new List<int>();
        for (int j = i; j >= 0; j--)
        {
            if (_list[j].StartsWith(pattern))
            {
                matchingIndexes.Add(j);
            }
            else
            {
                break;
            }
        }
        if (matchingIndexes.Count == GlobalVariables.LettersUsed.Count - item.Length)
        {
            foreach (var index in matchingIndexes)
            {
                _list.RemoveAt(index);
            }
            _list.Add(pattern);
            i = matchingIndexes.Last() - 1;
        }
    }

#endregion

Sort();
}

private void Sort()
{
    var listDividedByLength = new List<string>();
    for (int i = 1; i <= GlobalVariables.LettersToAdd + 2; i++)
    {
        var listCurrentLength = _list.Where(item => item.Length == i).ToList();
        listCurrentLength.Sort();
        listDividedByLength.AddRange(listCurrentLength);
    }
    _list = listDividedByLength;
}

public IEnumerator<string> GetEnumerator()
{
    return _list.GetEnumerator();
}
```

## Ciencias de la Educación y Humanidades

```

IEnumerator IEnumerable.GetEnumerator()
{
    return GetEnumerator();
}
}
}

```

El proceso de búsqueda mediante el algoritmo de encadenamiento de trayectorias requiere de una solución de trabajo y una solución guía de alta calidad.

### CONCLUSIONES

- El método de Path Relinking o método de re-encadenamiento de trayectorias es un importantísimo programa, ya que encuentra la ruta más corta generando los costos mínimos.
- Es un programa basado en la búsqueda de nuevas rutas cada vez mejorando las anteriores.
- El método presenta la capacidad de explorar sub-espacios de muy buena calidad que estén cerca a otras soluciones élites ya identificadas.
- El método de Path Relinking es un método programado para optimizar los costos de una forma impresionante, haciendo recorridos por distintas rutas y mejorando las ya existentes.

### REFERENCIAS BIBLIOGRÁFICAS

Bazaraa, M., Jarvis, J., Sherali, H. (2010). Linear Programming and Network Flows. Canadá: Wiley

Duarte, Martí, R., Campos V., Resende M., (2012), Extraído el 13 de septiembre de 2015 desde <http://www.escet.urjc.es/~aduarte/proceedings/maeb12b.pdf>

Gallego, R., Escobar A., Toro E., Técnicas Metaheurísticas de Optimización. Taller de publicaciones Universidad Tecnológica de Pereira 2008.

Rodríguez, C., (2010), Algoritmos heurísticos y metaheurísticos para el problema de localización de regeneradores. Tomado el 12 de septiembre de 2015 desde <https://ciencia.urjc.es/bitstream/handle/10115/4129/memoriaPFC%20Carlos%20Rodr%C3%ADguez.pdf?sequence=1&isAllowed=y>

Santamaría, J., (2004) Extraído el 7 de octubre de 2015 desde <http://www.redheur.org/sites/default/files/metodos/PR01.pdf>

Taha, H., (2004). Investigación de Operaciones. México: Pearson

Path Relinking Extraído el 7 de octubre de 2015 desde

<https://ccc.inaoep.mx/~emorales/Cursos/Busqueda/node97.html>.