

# Tecnología para Reciclar Tecnología: Un Plotter Ecológico

Mariano Scasso  
Dpto. de Ing. Elec. y de Computadoras  
Universidad Nacional del Sur  
Avenida Alem 1253 - Bahía Blanca, Buenos Aires

mariano.scasso@uns.edu.ar

**Resumen** El siguiente artículo describe la realización de una herramienta de dibujo denominada Plotter a partir de elementos reciclados. Para su control se utiliza un sistema embebido que interactúa con una PC a partir de un software diseñado especialmente.

## 1. Introducción

En la actualidad, y como consecuencia del rápido avance de la tecnología, existe una nueva problemática relacionada con la denominada “chatarra electrónica”, compuesta por dispositivos electrónicos de descarte. En consecuencia es común encontrarse con campañas para recolectar la chatarra y depositarla en lugares seguros, ya que muchos de los elementos que la componen son contaminantes. Sin embargo la tarea de buscar una nueva función a estos dispositivos y adaptarlos para que sean compatibles con las nuevas tecnologías no es sencilla [1]. Reciclar las diferentes partes que componen una máquina en desuso sería un paso hacia la disminución de basura, ya que en su interior se encuentran valiosos componentes, tales como motores, fuentes de alimentación, sistemas mecánicos, etc. Sin embargo las entidades ecologistas no cuentan con el conocimiento ni los recursos necesarios para realizar esas tareas [2].

Este proyecto trata sobre el diseño de un Plotter, entendiendo por esa denominación a una máquina capaz de dibujar o realizar mecanizados sobre un área plana, que no necesariamente debe ser papel (a diferencia de una impresora). El diseño a *plotear* se realiza con un software de PC. Luego la información se codifica y se envía al Plotter a través del puerto serie, donde es interpretada por un microcontrolador. Este último se encarga de decodificarla y accionar los dispositivos mecánicos necesarios para realizar el dibujo sobre el área de trabajo.

La motivación central es crear una herramienta con fines didácticos a partir de elementos mecánicos y electrónicos obtenidos de dicha chatarra, pensando en la aplicación futura para realizar circuitos impresos.

## 1.1. Descripción del problema

La construcción del Plotter involucra tres problemáticas complementarias: el desarrollo mecánico del dispositivo; el del sistema embebido de control y la interfaz gráfica con el usuario.

En primer lugar, se tiene el sistema mecánico del Plotter, que será conformado mediante partes extraídas de scanners e impresoras, tales como: engranajes, ejes, y motores paso a paso. Este último es un elemento fundamental de cualquier sistema electro-mecánico que requiera gran precisión, ya que avanza una determinada posición angular de forma perfectamente controlada.

Para controlar el dispositivo es necesario implementar un sistema embebido compuesto por un microcontrolador (AT89C51ED2 de ATMEL) y algunos circuitos electrónicos (amplificador de potencia, fuente de alimentación, circuitos de señalización y sensado). Su función es actuar como interfaz entre el Plotter y la PC. El microcontrolador se ocupa de controlar la rotación de los motores, detectar errores, establecer comunicación con la PC, etc.

Por último se requiere de un software para PC que cumpla la función de interfaz entre el usuario y la herramienta, y que permita a su vez realizar dibujos o importar los realizados con otro software. En este sentido el enfoque se centró sobre la compatibilidad con códigos de CNC (Control Numérico Computarizado) por ser el lenguaje utilizado mayormente por los sistemas CAD/CAM (Computer Assisted Desing/Computer Assisted Manufactured) industriales. Existen varias versiones de CNC, para este proyecto se optó por utilizar el g-code en su versión RS-274X por ser la más difundida [3]. Adicionalmente es deseable que la aplicación de software sea capaz de optimizar la ejecución del trabajo, ordenando de la manera más eficiente las figuras a dibujar. Con ese objetivo se trabajó sobre una herramienta que utiliza algoritmos genéticos para resolver una variante del “problema del viajante” (TSP). Además esta aplicación permite la realización de simulaciones en pantalla, como se describirá apropiadamente más adelante.

## 2. Sistema mecánico

La Figura 1 muestra un esquema que señala las principales partes del Plotter. El dispositivo consta de una base que debe ser capaz de desplazarse en dos direcciones, y a su vez es necesario un tercer eje encargado de comandar verticalmente el lápiz.

Debido a que se requieren movimientos de gran precisión, los motores encargados de desplazar la base son paso a paso. Este tipo de motores tiene la capacidad de rotar una cantidad fija de grados por cada paso, comúnmente  $7^\circ$ . A su vez cada paso ocurre cuando las bobinas del motor se energizan de una manera específica, permitiendo incluso desarrollar torque en una posición estática (mantener el motor frenado).

El dibujo se realiza mediante el desplazamiento de la base en las direcciones X - Y. Para transformar el movimiento de rotación de los motores en una traslación lineal se requiere un sistema mecánico, cuyo diseño es determinante a la hora

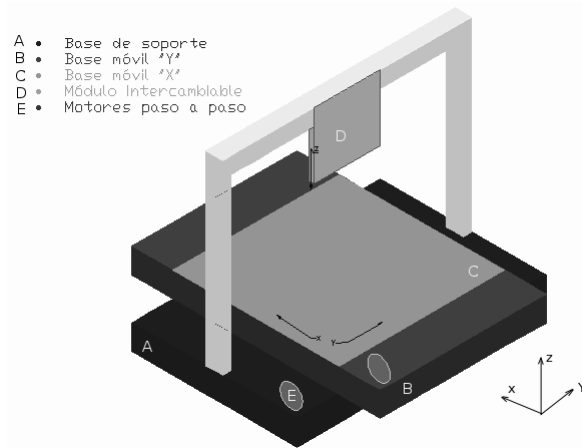


Figura 1: Esquema de las partes genéricas de un Plotter.

de establecer la precisión de la máquina (Figura 2). Por cada paso realizado por los motores la plataforma de dibujo se desplaza en una magnitud fija. En consecuencia es posible realizar movimientos exactos en 8 direcciones posibles como se explica en la Figura 3a. Cuando se requiere un movimiento en una dirección diferente a las indicadas se resuelve aproximándolo con la combinación de dos de movimientos básicos.

Si bien hay múltiples maneras de conseguir los desplazamientos necesarios, el prototipo aquí presentado está fabricado íntegramente utilizando chatarra electrónica. Puntualmente las bases móviles están fabricadas con las partes mecánicas de dos scanners completos colocados uno sobre el otro. El portalápiz se fabricó utilizando el sistema móvil del láser de una lectora de CD.

La expuesta es sólo una forma de construir el sistema mecánico. Cada diseño diferente afectará sensiblemente variables del dispositivo tales como la máxima área de trabajo, el paso de avance mínimo (resolución), el torque requerido a los motores, etc. Por lo anterior resulta fundamental que el sistema embebido encargado de comandar la máquina sea lo suficientemente versátil y genérico como para poder adaptarse en forma relativamente sencilla a las diferentes variantes mecánicas.

### 3. Sistema Embebido

En la sección anterior fueron descriptos los lineamientos generales del sistema mecánico del Plotter. Para accionar los diferentes mecanismos se requieren una serie de dispositivos electrónicos que se comandan utilizando el microcontrolador 89C51ED2 de Atmel. Esos dispositivos electrónicos, junto con el microcontrolador, constituyen un sistema embebido que hará de interfaz entre los elementos mecánicos del Plotter y el software de usuario.

Los requerimientos del sistema embebido se detallan a continuación:

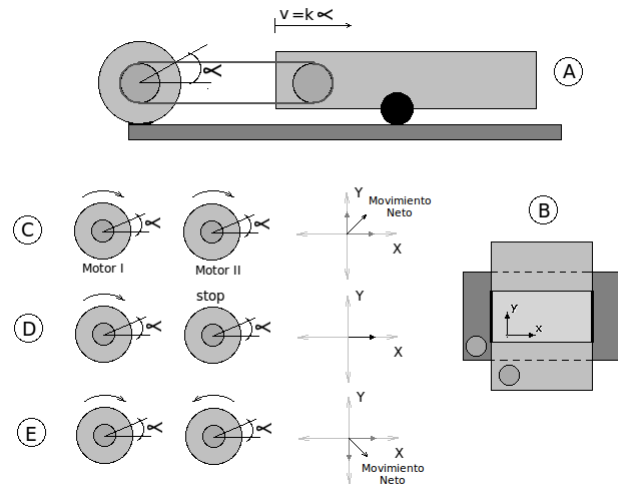


Figura 2: (A) Esquema representativo del sistema de engranajes y correas encargado de convertir el movimiento de rotación de los motores en una traslación lineal.

- Controlar el desplazamiento horizontal de la base de dibujo mediante el accionamiento de los motores paso a paso.
- Accionar el Lápiz (control de ascenso y descenso).
- Detectar errores de ejecución: sensado de finales de carrera.
- Indicar al usuario a través de un display LCD.
- Leer de un teclado simple para permitir el funcionamiento en modo manual y efectuar configuraciones básicas [5].
- Comunicarse con la PC.

El microcontrolador fue programado en lenguaje C utilizando Keil uVision3 como plataforma de desarrollo. Para realizar el sistema se utilizaron las siguientes características del microcontrolador:

- Cuatro puertos I/O:
  - Puerto 0: se utilizó como puerto de salida, se conecta a una etapa amplificadora y mediante esta se energiza las bobinas de los dos motores paso a paso destinándose cuatro bits por motor.
  - Puerto 1: Lectura de teclado.
  - Puerto 2 y 3: Tienen funciones compartidas. La parte baja de ambos puertos se utiliza para el control de los displays mientras que la parte alta se utiliza para monitorear los finales de carrera.
- Servicio de PWM (modulación por ancho de pulso) mediante la unidad PCA del microcontrolador: Se utilizan los pines 4 y 5 del Puerto 1 para generar la modulación necesaria para el control del motor de CC del lápiz. La modulación se controla por software en tiempo de ejecución.

- Interfaz serie: El AT89C51ED2 trae integrada una interfaz para puerto serie que es operada en forma asincrónica y por interrupciones para comunicarse con la PC.

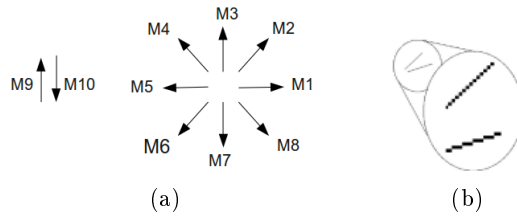
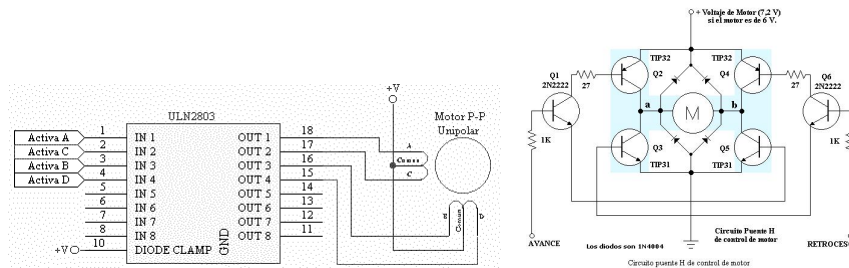


Figura 3: (a) Posibles direcciones de desplazamiento del Plotter, los movimientos M9 y M10 representan el descenso y ascenso del lápiz. (b) Zoom realizado sobre el dibujo de dos líneas en mapa de bits.

### 3.1. Control de los Motores Paso a Paso

La forma de accionamiento de esta clase de motores varía según su diseño constructivo. En general, constan de un rotor sobre el cual se monta un imán permanente, y un estator compuesto de una serie de bobinas. Cuando una bobina se polariza se produce un campo magnético inducido por la circulación de corriente. Este último interactúa con el campo magnético del imán permanente en el rotor ocasionándole un torque magnético que tiende a mantener alineados ambos campos. Por este principio es posible modificar la posición del rotor modificando la forma en que las bobinas son energizadas.



(a) Circuito adaptador de potencia entre el microcontrolador y los motores paso a paso.

(b) Circuito puente H

Figura 4: Circuitos

Como ya se mencionó la secuencia de rotación para cada motor es generada por el microcontrolador y enviada a través del Puerto 0. Debido a que los requerimientos de tensión y corriente de los motores no pueden ser satisfechos por el microcontrolador se coloca entre ambos una interfaz de potencia. La misma fue realizada utilizando el circuito integrado UNL2003 y el diagrama esquemático puede observarse en la Figura 4a.

### 3.2. Accionamiento del Lápiz

Como se explicó en la sección 2 el “portalápiz” se fabricó utilizando la parte mecánica de la bandeja de una lectora de CDs. Esta se acciona utilizando un motor de corriente continua, cuyo movimiento de rotación se transforma mediante una corredera en movimiento lineal. Dos finales de carrera colocados en la parte superior e inferior del recorrido permiten detectar si el lápiz se encuentra abajo (es decir el dispositivo esta dibujando) o arriba (la base se esta desplazando a una nueva posición). Cuando el Plotter recibe la instrucción de ascender o descender el lápiz, el motor de CC gira en uno de los dos sentidos hasta que el sensor pertinente se activa indicando que se alcanzó en la posición deseada.

Nuevamente, no es posible controlar el motor directamente desde el microcontrolador debido al exceso de potencia requerida, por lo cual se optó por la utilización de un driver electrónico denominado “Puente H”. Para modificar el sentido de giro de un motor de CC se requiere invertir la dirección del flujo magnético a través de sus bobinas, lo cual se consigue intercambiando la polarización de su fuente de alimentación. Esta es justamente la tarea desempeñada por el circuito indicando en la Figura 4b.

Para obtener un posicionamiento preciso y rápido es deseable que el motor acelere rápidamente y disminuya su velocidad al acercarse a la posición final. Este efecto se consigue comandando al Puente H mediante una señal PWM (Pulse width modulation). La modulación por ancho de pulso consiste en utilizar una señal de tipo onda cuadrada en lugar de un valor continuo de tensión. De esta forma al variar el ciclo de trabajo (proporción entre el tiempo en nivel alto y nivel bajo de la señal) se modifica el valor medio de tensión sobre el motor y consecuentemente su velocidad de giro. Como se indico al comienzo de la sección las dos señales PWM, para el ascenso y el descenso del lápiz, se generan con el microcontrolador utilizando los bits 6 y 7 del Puerto 1.

## 4. Software para PC

El software de PC desarrollado para actuar como interfaz entre el usuario y el Plotter se realizó utilizando Java como plataforma y fue nombrado JPlotterController. El programa en su grado de desarrollo actual ofrece las siguientes funcionalidades:

- Permite realizar dibujos conformados por figuras geométricas.
- Actúa como controlador del dispositivo implementando un protocolo de comunicación con el microcontrolador a través de un puerto serie RS-232.

- Permite optimizar el trabajo para que luego de ejecute por el Plotter en un menor tiempo.
- Realiza la simulación en pantalla del trabajo.
- Contempla la importación y exportación en el código de CNC RS-274X .

A continuación se describen los aspectos más destacables de la aplicación.

#### 4.1. Interfaz con el usuario y herramienta de dibujo

Debido a que transformarse en una herramienta de dibujo no es el objetivo principal de la aplicación, JPlotterController permite realizar dibujos simples ingresando los tipos básicos de figuras sobre un sistema de coordenadas XY. Para dibujos de mayor complejidad existen una gran cantidad de herramientas de diseño CAD en el mercado, que en su mayoría son compatibles con el formato RS-274X. De esta manera es posible importar un proyecto, ajustarlo y realizarlo con esta herramienta. La Figura 5a muestra la pantalla principal.

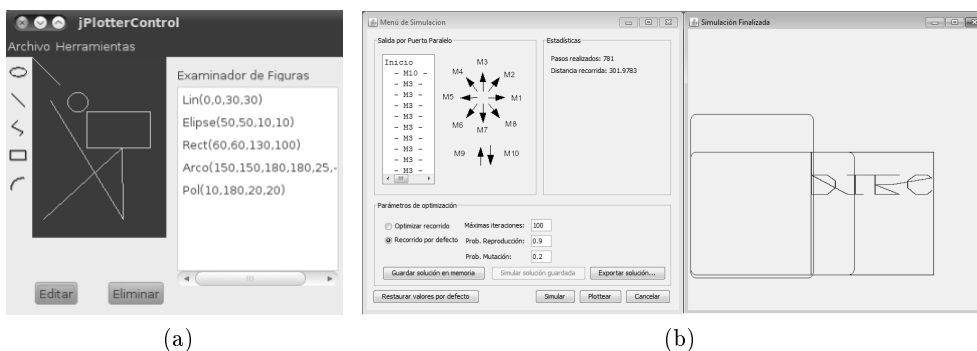


Figura 5: Capturas de pantalla: (a) Menú principal de JPlotterController. (b) Menú de simulación.

Si bien por una cuestión de extensión no es posible realizar una descripción detallada del software desde el punto de vista de la programación, es interesante explicar la forma en que se compone y representa el dibujo realizado por el usuario. Se denomina “área de trabajo” al lienzo donde se realiza el dibujo. Por cada figura que se agrega en el área de trabajo se crea una nueva instancia de un objeto que tiene una serie de propiedades relacionadas con la forma en que debe ser dibujado por el Plotter y su representación gráfica. Esta última se consigue modificando mediante herencia la forma en que se dibuja un objeto “Label” dentro del formulario principal. El diagrama de clases de la figura 6 aclara estos conceptos. Se observa que una figura tiene una representación simbólica que incluye una serie de atributos, entre ellos un punto de referencia y un punto final (Ref y Ptofinal). Estos últimos se utilizan para ubicar espacialmente cada

figura dentro del dibujo. El atributo “Imagen” esta asociado a un objeto del tipo “Img” el cual compone la representación gráfica de la figura en la pantalla. El método “Dibujar” es abstracto para la clase ancestro *Figura*, y luego es heredado y definido por cada tipo particular, ya que no es lo mismo dibujar un círculo que un rectángulo. Este método no se refiere al dibujo en la pantalla de la PC, sino a la acción del trazado del dibujo con el Plotter y cumple, entre otros aspectos, con la función de adaptar el diseño para que pueda ser efectuado.

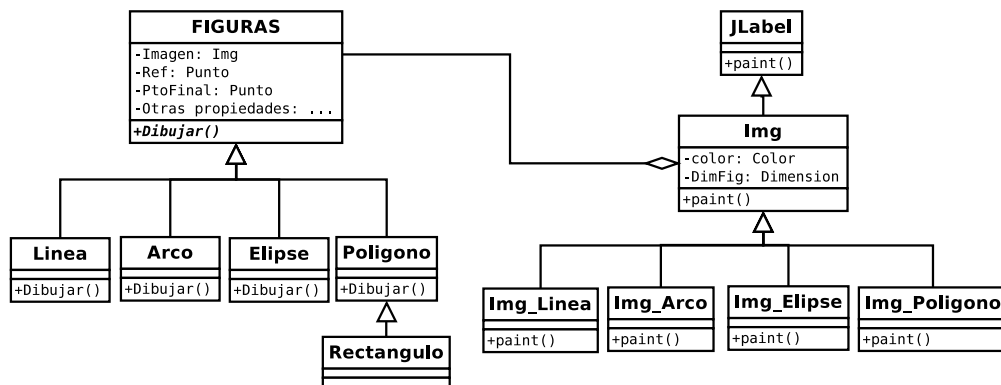


Figura 6: Diagramas UML básicos que indican la relación entre las “figuras” y su representación gráfica.

#### 4.2. Comando del dispositivo y microcódigo de comunicación

Para establecer la comunicación entre la PC y el Plotter es necesario un dialogo entre JPlotterController y el programa embebido en el microcontrolador.

Como muestra la Figura 3a la máquina es capaz de realizar un paso en 8 direcciones posibles, y cada una de ellas implica una combinación diferente de la rotación de los motores. En consecuencia las líneas que coincidan con las direcciones indicadas podrán ser dibujadas con exactitud, pero aquellas que no lo hagan deberán aproximarse. Este proceso es similar al utilizado por los programas de dibujo en el formato mapa de píxeles, en los cuales las líneas con una inclinación diferente a 45° se dibujan en forma aproximada (Figura 3b).

Debido a las dificultades mencionadas es necesario discretizar el dibujo y transformarlo en un conjunto de pequeñas líneas que puedan dibujarse con los desplazamientos básicos. Este proceso se ilustra en la Figura 7. En primer lugar se transforma cada una de las figuras dentro del área de trabajo en un conjunto de líneas y luego cada una de ellas se reduce a un conjunto de pasos en las direcciones indicadas en la Figura 3a utilizando el algoritmo de Bresenham [4]. Este algoritmo permite aproximar una línea mediante una cantidad determinada de movimientos discretos. Luego de realizar una codificación y empaquetado



para optimizar el proceso de transmisión, la información se envía a través del puerto serie al microcontrolador que se encarga de interpretar y ejecutar cada instrucción.

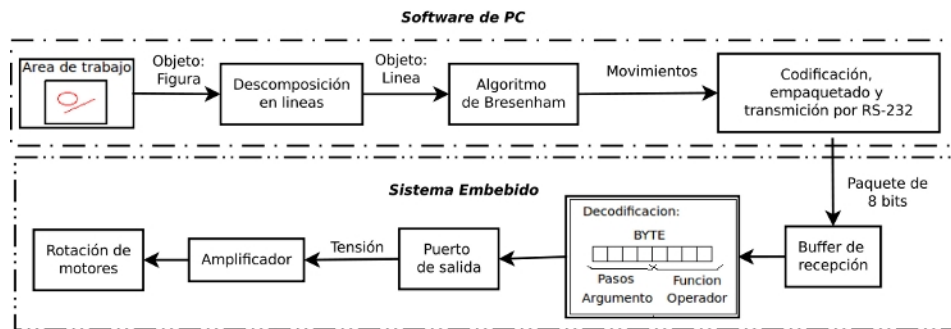


Figura 7: El esquema del proceso de dibujo

### 4.3. Optimización, solución de camino mínimo

A la hora de realizar el dibujo sobre el medio físico se utiliza el método indicado en la sección anterior, por lo tanto el dispositivo dibujará cada figura en el orden en que fueron insertadas. La figura 8a muestra que en determinadas situaciones se generan operaciones innecesarias.

#### 4.3.1. Problema del viajante (Travelling Salesman Problem)

El conocido TSP describe el problema de un vendedor que debe recorrer una serie de ciudades y quiere encontrar el camino que minimice la distancia a transitar.

La realización de un dibujo con el Plotter se trata de el mismo problema al cual se le agrega una restricción. Si pensamos el dibujo como una serie de nodos que deben ser conectados para conformar líneas (recordar que cualquier figura se aproxima por una cantidad de líneas), entonces es deseable pasar por cada uno de ellos recorriendo la distancia mínima, con la restricción de que deben mantenerse agrupados aquellos que conforman el inicio y el final de una línea.

Si bien el problema parece simple, no existe una forma analítica de hallar la solución. La única opción es probar todas las posibilidades hasta encontrar la mejor y el orden de tiempo de ejecución que se requiere para esto es muy alto. La manera más eficiente de resolver este tipo de problemas es mediante un algoritmo heurístico de optimización, como el Algoritmo Genético (GA), el cual reduce los tiempo de cómputo y permite obtener una solución aceptable.

#### 4.3.2. El Algoritmo Genético

El GA se caracteriza por tener un comportamiento basado en la recombinación

de los cromosomas y la evolución en los seres vivos. El proceso general es el siguiente:

**Inicialización:** Se comienza generando una cantidad  $N$  de soluciones (cromosomas) aleatorias llamado población, se calcula la aptitud de cada solución (la aptitud es la función a optimizar, en este caso la distancia total recorrida) y en base al valor de esta se ordenan los cromosomas.

**Generación:** Se genera una nueva población de cromosomas tomando como referencia a la población actual. Para esto se repite el siguiente proceso hasta obtener  $N$  nuevas soluciones.

- **Eliteísmo:** La solución con mejor aptitud pasa directamente a la nueva población sin alteración alguna. Este paso es un agregado que permite conservar la mejor solución obtenida.
- **Selección:** Se elijen dos soluciones en base a su aptitud (aquellos cromosomas con mejor aptitud tienen mayor probabilidad de ser seleccionados).
- **Recombinación:** Las soluciones elegidas (padres) pasan por un proceso de entrecruzamiento de donde se obtienen dos cromosomas hijos. La forma en la que se realiza el entrecruzamiento varía según la implementación pero básicamente consta de un intercambio de información de la solución contenida en los cromosomas seleccionados.
- **Mutación:** Por lo general este paso se toma en cuenta con una pequeña probabilidad, y permite que la solución contenida en alguno de los cromosomas hijos se vea alterada ligeramente.

**Reemplazo:** La nueva generación de cromosomas reemplaza a la anterior y nuevamente es ordenada. Luego se comprueba si se verifica la condición de finalización (la cual depende de la implementación), si se cumple, se continúa con el proceso de Finalización, sino, se retorna al de Generación.

**Finalización:** La solución resultante es aquella que posee la mejor aptitud.

#### 4.3.3. Restricciones al problema

Las soluciones al TSP se presentan en forma de un grafo dirigido o una lista de nodos denominado “ciclo hamiltoniano”. Este tiene la propiedad de que al intercambiar dos elementos cualesquiera en un ciclo, se obtiene uno nuevo que también es solución del TSP. Esto es muy ventajoso para cualquier algoritmo heurístico, sin embargo debido a las restricciones que se mencionaron anteriormente, no es posible intercambiar dos elementos cualesquiera. De modo que se optó por presentar las soluciones en forma de un árbol binario, donde aquellos nodos que estén bajo una restricción forman una rama secundaria con el primer nodo como cabecera, y éstos, junto a los nodos que no están dentro de una restricción, conforman la rama principal (figura 8b). Luego para seguir el recorrido basta con acceder al árbol en preorden. Si se desea alterar el mismo, solamente es posible hacerlo permutando los nodos de una misma rama (principal o secundaria) a la vez.

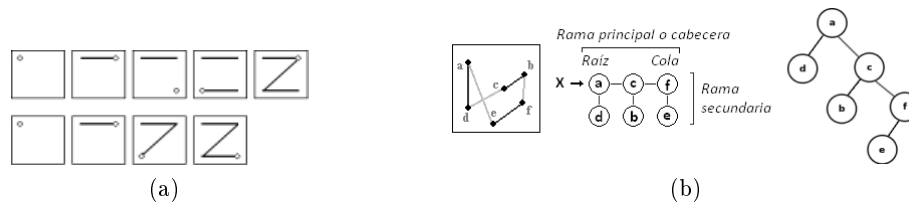


Figura 8: (a) Comparación entre la cantidad de operaciones necesarias para realizar un dibujo. (b) Solución presentada como árbol para un TSP con restricciones.

#### 4.4. Simulación en pantalla

Uno de los aspectos principales de JPlotterController es la posibilidad de realizar una simulación en pantalla previa a la realización del dibujo con el Plotter. Esto permite obtener información acerca de la cantidad de operaciones que demandará la ejecución y analizar la mejora del rendimiento luego de la optimización.

La simulación se realiza utilizando los mismos métodos que durante una ejecución real, solo que en lugar de transmitirse la información al Plotter la misma se almacena dentro de una matriz a partir de la cual se genera la animación.

Realizar efectos de animación en Java no es sencillo y detallar los métodos que permiten generarla escapan a este artículo. En forma breve, para generar la ilusión de movimiento se modifica mediante herencia el método “paint” de un JPanel. Este método describe la forma en que el panel se dibuja en la pantalla. El método re-paint() es llamado por el sistema cada un determinado tiempo para refrescar la ventana. De esta forma cada vez que el método re-paint() es ejecutado se dibuja un nuevo cuadro de la animación utilizando la información almacenada en una matriz. Para sincronizar la animación se utiliza un segundo thread de ejecución que refresca la pantalla de simulación en intervalos de tiempo fijos, consiguiendo así la ilusión de movimiento constante. La figura 5b muestra la pantalla de simulación.

#### 4.5. Compatibilidad con formato Gerber RS-274X

Como se mencionó al comienzo de la sección, el software de control del Plotter es compatible con el formato Gerber RS-274X. Esta característica permite que diseños complejos puedan realizarse utilizando una aplicación específica de dibujo para luego ser ejecutados por el sistema. Además, desde un punto de vista educativo, abre la posibilidad de utilizar esta herramienta (cuyo costo es muy inferior a las de uso profesional) para introducir al alumno en las técnicas básicas relacionadas con el CNC.

Nuevamente por cuestiones de extensión no es posible detallar el proceso por el cual se desarrolla esta función. Fundamentalmente se basa en aprovechar la

característica de que, tanto el formato el G-code utilizado comercialmente [11], como el microcódigo de operación del Plotter se tratan de lenguajes interpretados. En consecuencia el problema se reduce a crear un sistema que traduzca las operaciones en ambos sentidos.

## 5. Conclusiones

A lo largo de las secciones anteriores se describieron los aspectos fundamentales del desarrollo de un Plotter. Si bien el diseño presentado está orientado a la realización de dibujos en dos dimensiones, puede utilizarse como base para un dispositivo mas avanzado.

La precisión alcanzada por el prototipo actual es más que aceptable teniendo en cuenta que se fabricó en su mayor parte con elementos reciclados.

Su rango de aplicación es muy amplio, ya que con pequeñas modificaciones es posible realizar tareas diferentes. Tal sería el caso de intercambiar el lápiz de dibujo por un brazo robótico, obteniendo una herramienta de posicionamiento de precisión. A su vez la compatibilidad con CNC posibilita la aplicación del Plotter como un medio didáctico y de bajo costo para el sector educativo.

El software desarrollado en Java y la transmisión de datos vía RS-232 dota de portabilidad a todo el sistema, haciéndolo compatible con diferentes sistemas operativos y generaciones de PC.

La experiencia expuesta es una evidencia más de que es posible reutilizar la chatarra electrónica y darle nuevos usos productivos, reduciendo costos de implementación y brindando conciencia ecológica. En esta tarea los microcontroladores se transforman en una herramienta ideal por su versatilidad, robustez, y bajo costo.

## Referencias

1. Portal EduCar, Debate, "Basura electrónica, ¿Qué hacemos con ella?": <http://portal.educ.ar/debates/eid/tecnologia/debate/basura-electronica-que-hacemos.php>
2. Legislación de la Unión Europea, Directiva 2002/96/CE del Parlamento Europeo y del Consejo de 27 de enero de 2003: [http://europa.eu/legislation\\_summaries/environment/waste\\_management/l21210\\_es.htm](http://europa.eu/legislation_summaries/environment/waste_management/l21210_es.htm)
3. Wikipedia, Artículo G-code: [Ghttp://en.wikipedia.org/wiki/G-code](http://en.wikipedia.org/wiki/G-code)
4. .E. Bressenham, Algorithm for computer control of a digital plotter, IBM SYSTEMS JOURNAL, vol. 4, N° 1, 1965
5. Carlos Reynoso, Miguel Correia. "Sistema Embebido de Control, Monitoreo y Seguimiento Satelital Mediante SMS". Concurso de Trabajos Estudiantiles, JAIIO 2010 (Segundo Puesto), C.A.B.A., Argentina 2010.
6. Christian Nilsson - Heuristics for the Traveling Salesman Problem - Linköping University
7. Hiroaki Sengoku, Ikuo Yoshihara - A Fast TSP Solver Using GA on JAVA
8. Yingzi Wei, Kanfeng Gu2, Parallel Search Strategies for TSPs using a Greedy Genetic Algorithm, School of Information Science and Engineering, Shenyang Ligong University.

9. David S. Johnson, Lyle A. McGeoch, The Traveling Salesman Problem A Case Study in Local Optimization
10. Marek Obitko, Genetic Algorithms (1998), University of Applied Sciences, República Checa
11. Barco Gerber Systems Corporation, RS-274X Format User's Guide (1998)