

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA



**PROPUESTA DE MÉTODO DE CÁLCULO DE
CORRECCIÓN DE ALINEAMIENTO VERTICAL DE CADENAS
DE AISLADORES PARA LÍNEAS DE TRANSMISIÓN DE ALTA
TENSIÓN**

INFORME DE SUFICIENCIA

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ELECTRICISTA

PRESENTADO POR:

TULLIO AVELINO GIGLIO GONZALEZ

**PROMOCIÓN
2001 - II**

**LIMA – PERÚ
2006**

**PROPUESTA DE MÉTODO DE CÁLCULO DE CORRECCIÓN DE
ALINEAMIENTO VERTICAL DE CADENAS DE AISLADORES PARA LÍNEAS
DE TRANSMISIÓN DE ALTA TENSIÓN**

Dedico este trabajo a:
Mis padres, por darme la libertad de descubrir la vida
y por sus sabios consejos.
Mi familia, por estar siempre a mi lado y brindarme
el apoyo en mi carrera.
Y a Patty, por darle sentido a mi vida y
ser mi compañera para siempre.

SUMARIO

El presente trabajo plantea un método matemático exacto para el cálculo de la corrección de alineamiento vertical de cadenas de aisladores al momento del tendido y flechado del conductor; también conocido en el idioma inglés como “clipping offset”.

En el capítulo I se muestran las pautas y recomendaciones para el tendido de líneas de transmisión de alta tensión, basadas en la norma IEEE Std 524-1980 y traducidas del idioma inglés.

El capítulo II se refiere al planteamiento del problema, explicando claramente cómo se presenta este problema en el tendido de líneas de transmisión y la relación matemática de los momentos de fuerza que se presentan en la torre de transmisión.

El capítulo III trata de la demostración de las ecuaciones utilizadas para el cálculo propuesto. Todas estas ecuaciones están comprendidas dentro del cálculo mecánico de una línea de transmisión.

El capítulo IV plantea la solución del problema, basado en las ecuaciones del capítulo III y en algunas otras demostraciones matemáticas que se originan en los esfuerzos físicos en la cadena de aisladores.

El capítulo V trata del desarrollo de un programa aplicativo y su forma de utilizarlo y en el capítulo VI se muestra la solución de un caso práctico de la realidad nacional utilizando el programa aplicativo.

ÍNDICE

PROLOGO	1
CAPÍTULO I	
PAUTAS PARA EL TENDIDO DE LÍNEAS DE TRANSMISIÓN DE ALTA TENSIÓN	
1.1 Procedimientos de templado	2
1.1.1 Teorías de templado y corrimiento de cadena (Clipping Offset)	2
1.1.2 Formularios y registros	2
1.1.3 Criterio de diseño	2
1.1.4 Equipo	3
1.1.5 Lugar de tiro (esfuerzo) y la relación de la estructura de anclaje	3
1.1.6 Elevación del conductor	3
1.1.7 Longitud del tramo	3
1.1.8 Ubicación de los vanos	3
1.1.9 Cambio de tensión	4
1.1.10 Factores de control	4
1.1.11 Preparación previa de templados	5
1.1.12 Ejecución de la operación de templado	6
1.1.13 Técnicas para una satisfactoria revisión del desarrollo del flechado	8
1.1.14 Reacción del conductor a las tensiones de templado	9
1.2 Precauciones en las terminales	9
1.2.1 Esfuerzos eléctricos	9
1.2.2 Tensión y lugares de tiro	10
1.2.3 Estructuras terminales	10
1.3 Engrapado (clipping in)	10
1.4 Instalación del amortiguador (frenadores)	11
1.5 Instalación de espaciadores	11

CAPÍTULO II**PLANTEAMIENTO DEL PROBLEMA QUE SE PRESENTA EN EL TENDIDO DEL CONDUCTOR CUANDO LAS TORRES SE ENCUENTRAN A DESNIVEL**

2.1 Planteamiento del Problema	12
2.1.1 Antecedentes	12
2.1.2 Planteamiento	12

CAPÍTULO III**DEMOSTRACIÓN DE ECUACIONES UTILIZADAS**

3.1 Ecuación de la catenaria	15
3.2 Ecuación de la longitud para un cable a nivel	15
3.3 Ecuación de longitud para un cable a desnivel	16
3.4 Ecuación de desnivel	17
3.5 Ecuación de longitud para un cable a desnivel en función del desnivel	17
3.6 Ecuación de la flecha para un cable a desnivel	19
3.7 Ubicación cartesiana del punto medio y de los extremos del vano	21
3.8 Parámetro de la catenaria en función del tiro máximo	22

CAPÍTULO IV**PLANTEAMIENTO DEL MÉTODO DE SOLUCIÓN DEL CÁLCULO DE CORRECCIÓN DE ALINEAMIENTO VERTICAL DE CADENAS DE AISLADORES**

4.1 Definición de corrección de alineamiento vertical de cadena de aisladores	27
4.2 Procedimiento de cálculo	27

CAPÍTULO V**DESARROLLO DEL PROGRAMA APLICATIVO**

5.1 Software utilizado	35
5.2 Metodología utilizada	35
5.3 Manual de Usuario	36

CAPÍTULO VI**APLICACIÓN A UN CASO PRÁCTICO DE LA REALIDAD NACIONAL**

6.1 Datos de la línea de transmisión	42
6.2 Datos del tramo	42
6.3 Cálculo del corrimiento de cadena utilizando el programa aplicativo	43
CONCLUSIONES Y RECOMENDACIONES	45
ANEXO A	46
ANEXO B	50
BIBLIOGRAFÍA	100

PRÓLOGO

Hoy en día la energía eléctrica se ha convertido en necesaria e indispensable para el crecimiento económico de los países a través del desarrollo de sus industrias y tecnología. También se ha convertido en un servicio indispensable para la supervivencia y desarrollo de cada persona.

Es así que debido al aumento continuo de la población mundial, se necesita un aumento equivalente en las fuentes de energía eléctrica y en los medios de transporte de ésta. Por lo tanto será necesario la construcción de más líneas de transmisión para afrontar la demanda de energía futura.

Es conocido también que las fuentes de generación de energía hidroeléctrica se encuentran en su mayoría alejadas de las ciudades. Por lo tanto es necesario realizar el tendido de líneas de transmisión de alta tensión largas y en terrenos agrestes.

En el presente trabajo plantaremos un método de solución a uno de los tantos problemas de ingeniería que se presentan en el momento del tendido de una línea de transmisión, el problema del cálculo de corrección de alineamiento vertical de cadenas de aisladores. Éste cálculo trata de la corrección de la posición de la grapa que sostiene al conductor, cuando las diferentes torres de un tramo se encuentran a desnivel y las cadenas de aisladores con el conductor en las roldanas no quedan verticales. El cálculo se basa en que la longitud del cable en un tramo, cuando el cable está en las roldanas, es igual a la longitud del cable cuando el cable está en las grapas.

Para la demostración del método planteado, se ha desarrollado un programa aplicativo utilizando software libre de licencia (Open Source). Se ha utilizado la tecnología web con acceso a base de datos y para la programación se ha utilizado la metodología orientada a objetos.

Es necesario agradecer la colaboración de mi asesor, el Ing. Juan Bautista, por su ayuda incondicional y porque de sus investigaciones ha nacido la base teórica para la propuesta del método de cálculo del presente informe.

CAPÍTULO I

PAUTAS PARA EL TENDIDO DE LÍNEAS DE TRANSMISIÓN DE ALTA TENSIÓN

1.1 Procedimientos de templado

1.1.1 Teorías de templado y corrimiento de cadena (Clipping Offset)

Teóricamente, el templado del conductor esta basado en las funciones hiperbólicas describiendo una cierta curva catenaria. En la práctica, sin embargo, son frecuentemente utilizadas aproximaciones parabólicas de la catenaria.

La teoría del corrimiento de cadena (clipping offset) esta basada en el hecho que, entre estructuras de anclaje (tramos), la longitud total del conductor de flechado en las roldanas es igual a la longitud total del conductor de flechado en las grapas de suspensión.

El flechado y el corrimiento de cadena (clipping offset) están interrelacionados desde que las correcciones de flecha requeridas para cálculos de flechas son dependientes de los cálculos de corrimiento de cadena. La aplicación de flechas y clipping offset calculadas en esta forma producirán fuerzas horizontales balanceadas las cuales serán las mismas para cada estructura dentro del tramo.

1.1.2 Formularios y registros

Para ayudar en una recopilación exacta de datos de la sección de flecha, se debe preparar un conjunto de formularios diseñados para registrar exactamente todos los datos de campo, cálculos, etc., tan pronto como ellos sean obtenidos. Si surgen dudas mientras que el trabajo esta en proceso, la disponibilidad de registros podrían ayudar grandemente en proveer las respuestas.

1.1.3 Criterio de diseño

Un completo conjunto de criterios de diseño para la sección de flecha debe estar disponible en el campo, se debería incluir datos de diseño de estructura, datos de flechado, perfiles de

línea, conductor y plantillas de la línea de flecha de tiro, etc.

1.1.4 Equipo

El equipo requerido para el templado incluye teodolitos (o dispositivos visores similares), radios portátiles, termómetros de conductor, plataformas de templado y blancos (puntos de referencia), varillas de señalización, cintas para medir y dispositivos diversos para marcar.

1.1.5 Lugar de tiro (esfuerzo) y la relación de la estructura de anclaje

El lugar para realizar el tiro debe ser adyacente a la estructura de anclaje siempre que sea posible. Sin embargo si la estructura de anclaje es una estructura terminal, el lugar de tiro debe ser colocado varios vanos más allá. Cuando esto ocurre, el conductor entre el lugar de tiro y la estructura de anclaje debe estar aflojado hacia abajo tanto como sea posible en la terminal de la flecha para minimizar el pre-esfuerzo del conductor. Esta es una situación no deseable desde que la siguiente sección de flecha incluirá al conductor pre-esforzado (o pre-tensionado) con el conductor no tensionado. Tal situación debe ser evitada.

1.1.6 Elevación del conductor

Bajo ciertas condiciones, la elevación del conductor dentro de un tramo puede ocurrir a tiro de templado. Se requerirá mantener sujetos los bloques o rodillos (o cilindros) de elevación, o ambos, para mantener el conductor en las roldanas y así compensar estas condiciones.

1.1.7 Longitud del tramo

Un tramo no debe de exceder de 4 ½ millas (7 Km.), o aproximadamente 20 vanos, en longitud. Pueden ocurrir excepciones, pero deben de ser evitadas, particularmente en colinas o terrenos montañosos. Una excesiva sección de flecha usualmente ocasionará dificultades en el templado o flechado.

1.1.8 Ubicación de los vanos

Antes que los vanos sean seleccionados, un perfil a escala del tramo total debe ser revisada para proveer una completa figura de la relación entre el terreno y el conductor. Este perfil es una herramienta valiosa para ser usada en la selección de los vanos y puede enfatizar la localización de los problemas potenciales.

Los vanos de flechado deben estar en, o cerca, de cada terminal del tramo. Para secciones de flecha de más de 2 millas de largo, vano(s) de flechado adicionales cerca al centro del tramo deben ser utilizados. Estos vanos deben de ser los más largos y más nivelados. Si el vano no es un vano nivelado, es mejor si el teodolito es localizado en la estructura más baja desde donde el conductor de control es incrementado. Los vanos de flecha deben también ser colocados sobre cada lado de la línea en ángulos mayores a 10 grados.

1.1.9 Cambio de tensión

Los cambios de tensión pueden ocurrir en cualquier punto dentro de un tramo donde una estructura terminal es localizada. La situación más complicada ocurre, sin embargo cuando los cambios de tensión dividen la sección de flechado en 3 o más partes separadas, cada una de las cuales debe ser templadas independientemente del otro. Bajo estas condiciones, 2 o más vanos principales, y por tanto 2 o más tensiones requeridas, existen dentro de la sección de flechado. Aunque el conductor es continuo a través de la sección de flechado total, los cambios de tensión pueden ser establecidos mediante la terminación o el uso correcto de grapas y winches. Las estructuras terminales siempre existirán en cualquier punto donde estén los cambios de tensión del conductor, pero la mera existencia de la estructura terminal no siempre implica un cambio de tensión.

1.1.10 Factores de control

Cuando se escoge un método de templado, se debe de tener en cuenta que el punto de tangencia de la línea de vista del teodolito al conductor debe caer en el tercio mediano del vano. La referencia de los perfiles usualmente dará una indicación del mejor método de templado a utilizar. Por ejemplo, las estructuras altas sobre terrenos llanos y vanos cortos indican que los métodos mostrados en las figuras 1.2 y 1.3 probablemente proveerán el mejor control. En colinas, vanos largos y flechas grandes, indican que el método mostrado en la figura 1.1 podría ser el mejor. Después que los vanos de flechado hallan sido seleccionados, ellos deben ser revisados en campo por cualquier dificultad potencial la cual podría ocurrir durante el templado. Al mismo tiempo, los ejes de templado deben ser estabilizados si es requerido, y deben ser obtenidas las medidas requeridas para el cálculo de templado. Aunque el cronómetro de templado es algunas veces usado, la exactitud restringe su aplicación para conductores más pequeños y vanos más cortos que los normalmente hallados en la función de transmisión.

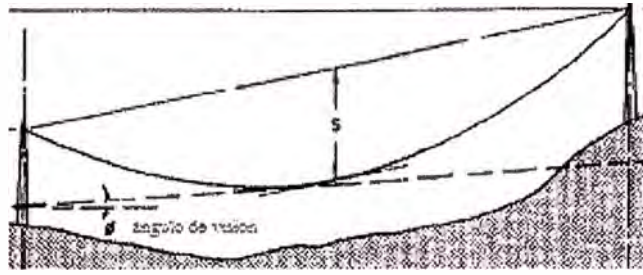


Fig. 1.1 Método de templado en vanos largos

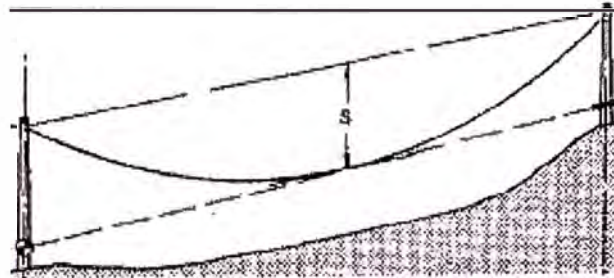


Fig. 1.2 Método de templado en vanos cortos

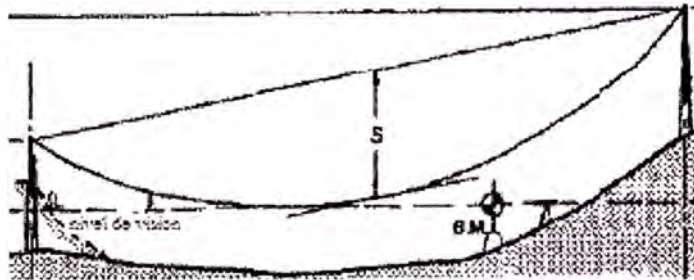


Fig. 1.3 Método de templado en vanos llanos y cortos

1.1.11 Preparación previa de templados

Las preparaciones para el funcionamiento de los mecanismos para el templado del conductor deben ser completadas bien en avance de la actual operación de templado. De otro modo, se puede incurrir en costos excesivos y retraso.

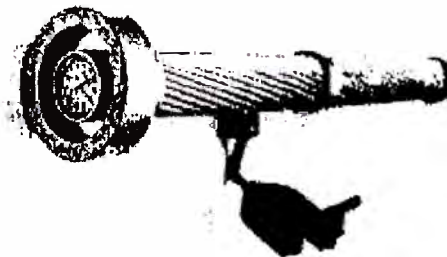


Fig. 1.4 Termómetro dentro de un contenedor

Cuando se es requerido, los ejes de teodolitos para los vanos de flecha deben ser colocados y apilados, las marcas de referencia de altura ubicados en las estructuras, y las plataformas de templado y ménsulas o soportes de montura para teodolitos deben ser instalados. Los termómetros de templado deben ser instalados con suficiente anterioridad a la operación de templado para permitir una estabilización de temperatura y deben estar montados lo suficientemente lejos de la tierra para evitar el efecto de la radiación del calentamiento de la tierra. Los termómetros deben ser insertados dentro de un contenedor (algunas veces en una sección del conductor) para representar la temperatura interna del conductor. (Ver figura 1.4) . Dos termómetros deben ser usados y deben ser colocados en los vanos de flecha cercanos, o en las terminales de las secciones de flecha.

La disponibilidad de suficientes radios portátiles debe estar asegurada y, si es necesario el transporte debe estar contratado.

Todo el personal de templado debe estar seguro de que ellos tienen el equipo apropiado y los datos de templado en el lugar. La persona quien controla el templado debe tener en su posesión un completo juego de registros pertinentes de toda la operación de templado.

Debido a las condiciones adversas del terreno, el personal de templado no siempre podrá observar todos los vanos de la sección de flecha. Un estudio del perfil de la sección de flecha normalmente revelará tal situación. Si la condición existe, una ayuda adicional será requerida para asegurar que los conductores estén equitativamente templados en los vanos ciegos.

1.1.12 Ejecución de la operación de templado

Después que todas las preparaciones hallan sido hechas y todo el personal asociado con la operación de templados esté en posición, la persona quien controla el flechado debe transmitir todos los detalles de último minuto al operador de tiro.

Él debe obtener las lecturas del termómetro de último minuto y usar el promedio de las 2 lecturas ajustadas para un incremento o decremento estimado en la temperatura a la terminación del flechado como la temperatura para el flechado del conductor. Esta información debe ser transmitida a todo el personal involucrado en el templado.

Nunca flechar un conductor al nivel de un conductor templado previamente. Flechar todos los conductores basados en el criterio de temperatura de diseño solamente. Al momento del

templado de los conductores, la flecha de cualquier fase dada debe estar dentro de las 6 pulgadas del valor teórico para las condiciones de temperatura existentes.

La flecha de todas las fases de un circuito debe tener tolerancias y direcciones similares a la flecha teórica. Los subconductores dentro de una fase deben tener tolerancias entre uno y otro de no más de 2 conductores de diámetros con un máximo de 2 pulgadas. Durante la revisión de las flechas un momento después del templado original, se debe recordar que el efecto CREEP incrementará la flecha y deben ser permitidos grandes límites de tolerancia para esto y otros valores inevitables. Aunque es deseable revisar las flechas tan pronto como sea posible, se debe recordar que pueden introducirse errores durante el engrapado y los procesos terminales.

Las comunicaciones y cooperación entre el personal que está realizando el templado y el operador de tiro, y entre el mismo personal de templado, son esenciales. El personal debe mantener al operador de tiro constantemente informado del movimiento del conductor, y si están siendo templados conductores adicionales, ellos lo deben mantener avisado del de la igualdad de espaciamiento que existe entre los conductores.

El conductor es templado en orden progresivo del extremo del tensionador del tramo hacia el winche de tiro. Por eso, en tanto el operador del tiro inicia el movimiento del conductor hacia el winche, cada persona en un vano de flecha, progresivamente del extremo de tiro al extremo del tensionador deben informar a la persona que está templando de los movimientos del conductor en tanto este se mueve a través del tramo. Dos beneficios son derivados mediante este método de comunicación. Primero, la persona que está templando, sabe cuando esperar el movimiento del conductor en su vano de flecha, y segundo, el operador de tiro sabe cuando debe disminuir o parar el tiro.

El flechado del conductor actual es iniciado mediante la persona que controla la flecha y es ejecutado primero por la persona en el vano de flecha más lejano del operador de tiro trabajando con personas que hallan empalmado el conductor en el vano conteniendo anclajes para las flechas previas. En tanto el conductor es lentamente soltado de los anclajes, la persona que está templando debe hacer que el operador de tiro afloje el conductor hasta que este ligeramente debajo de la flecha. Esta condición debe ser mantenida hasta que el conductor este completamente suelto de los carretes que lo sostienen. Una vez que el conductor este completamente suelto se puede realizar el tiro para el flechado. Si los conductores que están siendo tensados son conductores unidos. Ellos deben ser llevados a flecha lo más equivalentemente posible. Si uno de los

subconductores fuese inadvertidamente tensionado sobre la flecha en el vano de flecha, pueden desarrollarse dificultades serias. En esta situación, un intento para aflojar un subconductor de la atadura a flecha usualmente resulta en una desigualdad en todos los demás vanos. Si esta situación ocurriese después de un intento para aflojar un subconductor para flecha, el flechado debe ser detenido y la juntura (atadura) entera aflojada debajo de la flecha y nivelada. Entonces puede ser hecho otro ensayo para flechar conductores. Una vez que el primer vano de flecha halla sido llevado a flecha, los subconductores de la unión de conductores deben ser revisados para igualarlos, y entonces debe ser templado el siguiente vano de flecha. Las desigualdades en los vanos de flecha en el medio o en el extremo de tiro de una sección de flecha pueden ser usualmente corregidos mediante algunas manipulaciones de los conductores y bajo condiciones normales no resultarán en comenzar el flechado nuevamente.

Los ensayos para flechar un conductor en días de viento excesivo deben ser evitados desde que serios errores pueden resultar debido a la elevación del conductor causado por la presión del viento sobre el conductor. Si ocurriesen condiciones de viento severas después que un flechado en progreso, las tolerancias para la elevación del conductor deben ser exactas o el flechado debe ser detenido.

1.1.13 Técnicas para una satisfactoria revisión del desarrollo del flechado

Hay varias técnicas las cuales son empleadas para determinar si un conductor es correctamente templado (flechado). Como se estableció anteriormente, el conductor es templado progresivamente rumbo (o hacia) el extremo de tiro de una sección de flecha. En tanto el primer vano de flecha llega a la flecha la segunda persona a flechar debe hallar que el conductor en su vano de flecha esta muy alto. Esto tiende a esperarse y es normal a menos que el conductor este excesivamente alto. En tanto la segunda persona afloja su conductor a flecha y la tercera persona debe hallar el conductor demasiado bajo en su vano de flecha, y así respectivamente hasta que el flechado este completado. Si cualquiera de las personas que están templando no tienen las condiciones requeridas cuando el conductor es llevado a flecha en el vano de flecha precedente, las operaciones de templado deben ser detenidas hasta que el problema este localizado.

Si las condiciones anteriores son conocidas, es posible un desarrollo del flechado satisfactorio. Sin embargo si un intento de flechado en cualquier vano de flecha resulta en

serios movimientos de un vano de flecha previo, el problema es otra vez posible, y la operación de templado debe ser detenida hasta que el problema este localizado.

Cuando la flecha este completada, se debe registrar una lectura de tensión si se ha usado un dinamómetro. La lectura debe ser muy cercana a las tensiones nominales esperadas, el problema debe ser localizado y se deben hacer las correcciones antes que el flechado completo sea aceptado.

1.1.14 Reacción del conductor a las tensiones de templado

La reacción del conductor a las tensiones aplicadas durante las operaciones de templado es similar a la onda creada por la caída de una piedra en el agua. Una vez que la onda se ha iniciado, este continua por algún periodo del tiempo. Similarmente, cuando las tensiones son aplicadas al conductor en el extremo de tiro de una sección de flecha el movimiento del conductor es iniciado en aquel punto, y aunque la tensión pueda ser mantenida constante (paradas de tiro), el movimiento del conductor continúa hacia el otro extremo en un rango decreciente. Este movimiento debe ser tomado en cuenta durante el templado del conductor.

Las roldanas las cuales son usadas para el flechado del conductor no son de baja fricción y, por lo tanto, pueden causar problemas durante una operación de templado. Si una o más de las roldanas se llegan a atascar, el templado puede llegar a ser muy dificultoso. Una roldana la cual oscila en dirección del tiro puede ser una indicación de una roldana defectuosa. Si ocurriesen dificultades de templado inexplicados, las roldanas deben ser revisadas. Las tensiones aplicadas al conductor para vencer las roldanas pegadas o atascadas pueden causar un repentino y abrupto movimiento del conductor en los vanos de flecha y rápidamente causar pérdida de flecha, particularmente si el conductor, esta ya cerca de flecha.

1.2 Precauciones en las terminales

1.2.1 Esfuerzos eléctricos

Los esfuerzos eléctricos que existen cuando el trabajo de la terminal esta siendo ejecutado, son análogos a aquellos que existen durante las operaciones de empalme. Por lo tanto, se deben tomar precauciones para prevenir al personal de que tenga accidentes en serie con un circuito de potencial eléctrico.

1.2.2 Tensión y lugares de tiro

La continuidad de tierra y enlace (o ligadura) deben ser mantenidos cuando el conductor o líneas de tiro conductivas, o ambas, son transferidas entre piezas del equipo, o entre piezas de equipo y anclajes. En la mayoría de casos, este será necesario para mover una tierra existente sobre un conductor o línea de tiro antes que este pueda ser transferido. Antes de remover la tierra existente, la persona debe instalar su propia tierra personal para asegurar que el no se ubicara así mismo en serie a tierra con el conductor o línea a ser transferida.

1.2.3 Estructuras terminales

Previas a la instalación o remoción de barrenas terminales sobre una estructura de metal, deben ser instaladas tierras personales sobre los conductores en ambos lados del área de trabajo determinada y conectadas a la estructura. Si la estructura es de madera, ellos deben ser conectados a una fuente de tierra común. En algunos casos, después de que un extremo de la barrena halla sido permanentemente unido a un conductor, la inducción eléctrica puede ser tan severa que una tercera tierra personal será requerida para enlazar el extremo flojo de la barrena al otro conductor de manera que el conductor pueda ser permanentemente unido.

1.3 Engrapado (clipping in)

La porción de engrapado en la operación de fijar el conductor envuelve el trabajo posterior al flechado y el marcado vertical de los conductores. Esto implica la remoción de los conductores de las roldanas y su colocación en sus grapas de suspensión permanente, unidas a las cadenas de aisladores.

El engrapado empieza una vez que el conductor ha sido llevado a flecha y es iniciado mediante el emplazamiento de marcas verticales sobre el conductor directamente debajo de los puntos de unión del aislador sobre las estructuras con un mástil de marcado vertical. Este marcado es hecho tan pronto como sea posible después de alcanzada la flecha para minimizar el efecto CREEP y movimientos posibles del conductor entre vanos. En terrenos rugosos, los corrimientos de cadena (Clipping Offset) pueden ser usados por lo cual la grapa de suspensión, en vez de que sea colocada en la marca vertical, es corrida (Offset) una distancia calculada desde la marca vertical para compensar la desigualdad del terreno y para permitir a la cadena del aislador cuelgue verticalmente cuando todas las estructuras hallan sido engrapadas.

Cuando el engrapado esta siendo hecho, se debe tener cuidado para tener la certeza de que los conductores están conectados a tierra para el engrapado, a pesar del hecho de que las líneas están siendo engrapadas sin estar unidas a cualquier fuente eléctrica. Esto implica ubicar una tierra personal en el conductor del área que esta siendo trabajada. Después que los conductores hallan sido marcados, el personal levanta el peso de los conductores, permitiendo la remoción de las roldanas, y las grapas de suspensión, y la varilla del arnés son usadas, para ser ubicado sobre los conductores. El levantamiento es normalmente hecho mediante el uso de un winche suspendido de la estructura y el gancho del levantamiento del conductor el cual es diseñado para que no haga hendiduras o curvas o esfuerzos severos al conductor. Después de emplazar las grapas de suspensión sobre el conductor, los ganchos son bajados, emplazando de ese modo el peso del conductor sobre las grapas de suspensión y completando el montaje. Donde sean usados conductores unidos, los conductores múltiples pueden ser levantados simultáneamente mediante el uso de una yunta de los arreglos de soporte de los ganchos y un winche u otro medio de levantamiento.

1.4 Instalación del amortiguador (frenadores)

Los amortiguadores son normalmente ubicados sobre los conductores inmediatamente después del engrapado para prevenir cualquier posible daño por la vibración del viento a los conductores los cuales a tensiones criticas y condiciones de viento pueden ocurrir en materia de unas cuantas horas.

1.5 Instalación de espaciadores

Siguiendo la secuencia de operaciones de engrapado para líneas de conductor empalmado, los espaciadores deben ser usualmente instalados. Esto es hecho emplazando al personal sobre los conductores mediante el uso de un carro para el conductor para llevarlo de estructura en estructura. Dependiendo de la cantidad de línea a ser espaciada y el equipo disponible, los carros pueden ser accionados manualmente, remolcados por personas que están en tierra o en una estructura adyacente con cuerdas, o accionados por un pequeño motor sobre el carro mismo. El cuidado debe ser ejercido para asegurar que toda la carga concentrada del hombre, el carro y el equipo no incrementa la flecha lo suficientemente para causar una descarga sobre obstrucciones por la cual el carro pasará. La instalación de

los espaciadores varía con el tipo y la manufactura del espaciador y es normalmente echo de acuerdo a las recomendaciones del fabricante.

CAPITULO II

PLANTEAMIENTO DEL PROBLEMA QUE SE PRESENTA EN EL TENDIDO DEL CONDUCTOR CUANDO LAS TORRES SE ENCUENTRAN A DESNIVEL

2.1 Planteamiento del Problema

2.1.1 Antecedentes

En el pasado los diferentes métodos desarrollados para calcular las correcciones de flecha y corrimiento de cadena, incluyeron simplificaciones para permitir que los cálculos sean hechos manualmente. Ahora con todos los avances en el campo del calculo electrónico, podemos usar una aproximación que más adecuadamente representa la situación. El método presentado aquí trabaja con las ecuaciones exactas, sin utilizar aproximaciones de series infinitas.

2.1.2 Planteamiento

Para aclarar el problema utilizaremos el siguiente gráfico:

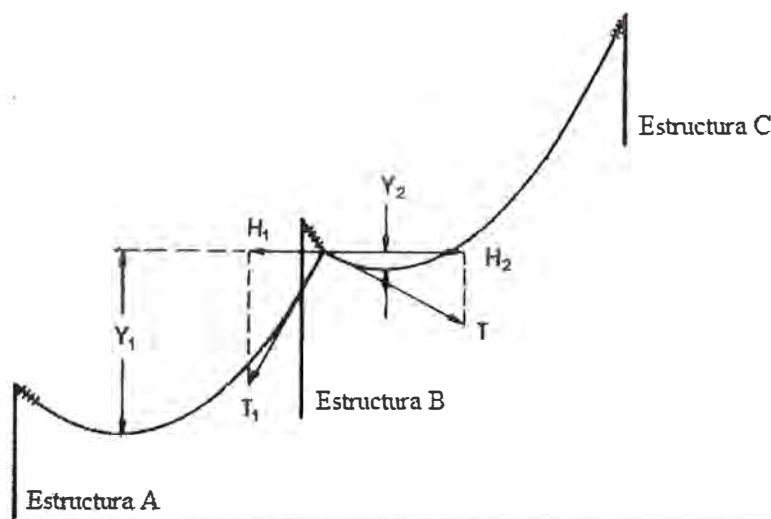


Fig. 2.1 Ilustración de las fuerzas horizontales desbalanceadas con el conductor en la roldana

La figura 2.1 incluye el diagrama de fuerza en una estructura intermedia con el conductor en una roldana. El conductor esta conectado en la estructura A, tensionada sobre una roldana en la estructura B, y con una terminal en la estructura C; la longitud correcta del conductor es conectada entre las 2 terminales.

Desde que el conductor esta en equilibrio en la estructura B; la tensión con el conductor a medida que este deja la roldana sobre un lado es igual a la tensión en el conductor a medida que este abandona la roldana en el otro lado. Por lo tanto el vector T_1 es igual al vector T_2 . Sin embargo, desde que los 2 vectores abandonan la roldana a diferentes ángulos, las componentes horizontales de estos 2 vectores, H_1 y H_2 no son iguales. La diferencia entre estos dos vectores horizontales causa que el aislador del cable quede colgado fuera de una posición vertical.

La magnitud de esta diferencia puede ser calculada como sigue:

$$T_1 = T_2 \quad (2.1)$$

De la ecuación básica de la catenaria,

$$T_1 = H_1 + wY_1 \quad (2.2)$$

$$T_2 = H_2 + wY_2 \quad (2.3)$$

Así:

$$H_2 - H_1 = w(Y_1 - Y_2) \quad (2.4)$$

Pero:

$$Y_1 - Y_2 = E_2 - E_1 \quad (2.5)$$

Por lo tanto:

$$H_2 - H_1 = w(E_2 - E_1) \quad (2.6)$$

en el cual:

$E_2 - E_1$ = Diferencia de los puntos más bajos de 2 catenarias adyacentes

$H_2 - H_1$ = Diferencia en las tensiones horizontales de 2 catenarias adyacentes.

w = Peso del conductor por unidad de longitud

Aún cuando la longitud total del conductor entre las 2 terminales es correcta, la distribución del conductor entre los 2 vanos no es correcta. Hay demasiado conductor en el vano inferior e insuficiente conductor en el vano superior. Para corregir esto e igualar las

tensiones horizontales, es necesario tensionar algún conductor del vano inferior dentro del vano superior.

En suma, para que estén correctamente la flecha y el engrapado de un conductor en unas series de tramos suspendidos que atraviesan una ladera inclinada escarpadamente (gran pendiente), es necesario hacer 2 cosas; primero, asegurar que la cantidad correcta del conductor es conectada entre las terminales. Este resultado es conseguido mediante el flechado del conductor para valores de flecha ajustados. Segundo, asegurar que el conductor este distribuido apropiadamente entre los vanos para igualar las tensiones horizontales. Sujetar las cadenas del aislador al conductor a las distancias del corrimiento, las cuales igualara estas tensiones.

En este trabajo plantearémos la solución al siguiente problema: “Encontrar el valor adecuado del corrimiento de la cadena de aisladores, para que ésta, quede perfectamente vertical luego de realizar el engrapado del conductor”.

CAPITULO III DEMOSTRACIÓN DE ECUACIONES UTILIZADAS

3.1 Ecuación de la catenaria

$$y = C \cosh\left(\frac{x}{C}\right) \quad (3.1)$$

3.2 Ecuación de la longitud para un cable a nivel

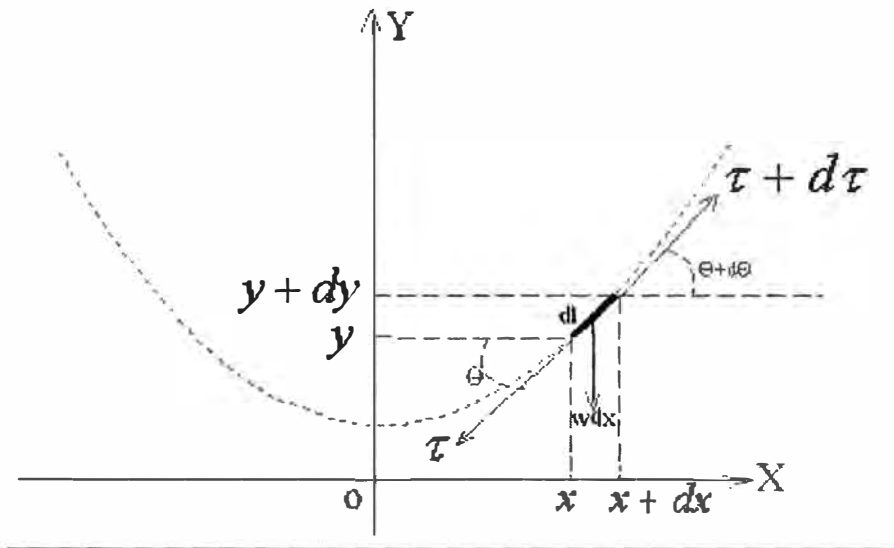


Fig. 3.1 Gráfica de la ecuación de la catenaria

Tomamos un diferencial de longitud del cable:

$$dl = \sqrt{(dx)^2 + (dy)^2} \quad (3.2)$$

De la ecuación (3.1), obtenemos:

$$dy = \sinh\left(\frac{x}{C}\right) dx \quad (3.3)$$

reemplazando la ecuación (3.3) en la ecuación (3.1)

$$dl = \sqrt{(dx)^2 + \left(\sinh\left(\frac{x}{C}\right) dx \right)^2}$$

$$dl = \left[\sqrt{1 + \left(\sinh\left(\frac{x}{C}\right) \right)^2} \right] dx \quad (3.4)$$

pero :

$$\cosh^2\left(\frac{x}{C}\right) - \sinh^2\left(\frac{x}{C}\right) = 1$$

reemplazando en (3.4):

$$dl = \cosh\left(\frac{x}{C}\right) dx \quad (3.5)$$

integrando para todo el vano entre $-\frac{a}{2}$ y $\frac{a}{2}$, tenemos:

$$L = \int_{-\frac{a}{2}}^{\frac{a}{2}} \cosh\left(\frac{x}{C}\right) dx$$

$$L = 2 \int_0^{\frac{a}{2}} \cosh\left(\frac{x}{C}\right) dx$$

$$L = 2C \sinh\left(\frac{a}{2C}\right) \quad (3.6)$$

3.3 Ecuación de longitud para un cable a desnivel

Necesitamos integrar la ecuación (3.5) entre la abscisa x_A y x_B :

$$\int_{x_A}^{x_B} \cosh\left(\frac{x}{C}\right) dx$$

de donde obtenemos:

$$L = C \left[\sinh\left(\frac{x_B}{C}\right) - \sinh\left(\frac{x_A}{C}\right) \right] \quad (3.7)$$

3.4 Ecuación de desnivel

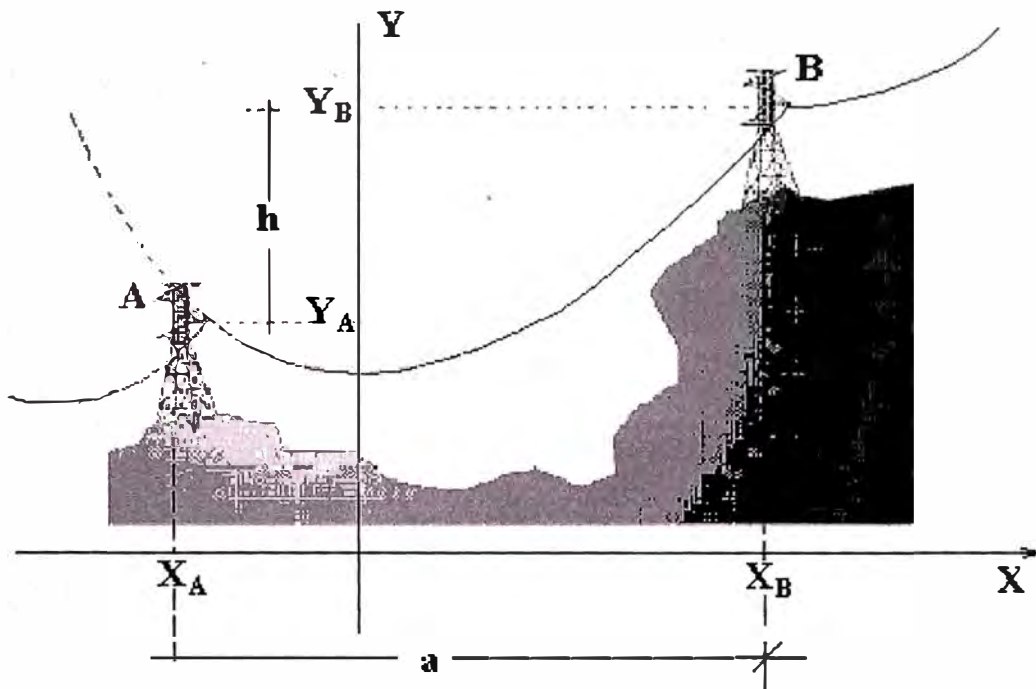


Fig. 3.2 Diferencia de alturas para dos torres a desnivel

De la figura 3.2:

$$h = y_B - y_A \quad (3.8)$$

reemplazando (3.1) en (3.7):

$$h = C \left[\cosh\left(\frac{x_B}{C}\right) - \cosh\left(\frac{x_A}{C}\right) \right] \quad (3.9)$$

3.5 Ecuación de longitud para un cable a desnivel en función del desnivel

Sumando y restando las ecuaciones (3.7) y (3.9) obtenemos:

$$L + h = C \left[\sinh\left(\frac{x_B}{C}\right) + \cosh\left(\frac{x_B}{C}\right) - \sinh\left(\frac{x_A}{C}\right) - \cosh\left(\frac{x_A}{C}\right) \right] \quad (3.10)$$

$$L - h = C \left[\sinh\left(\frac{x_B}{C}\right) - \cosh\left(\frac{x_B}{C}\right) - \sinh\left(\frac{x_A}{C}\right) + \cosh\left(\frac{x_A}{C}\right) \right] \quad (3.11)$$

sabemos que:

$$\cosh p + \sinh p = e^p$$

$$\cosh p - \sinh p = e^{-p}$$

entonces las ecuaciones (3.10) y (3.11) se convierten en:

$$L + h = C(e^{\frac{x_B}{C}} - e^{\frac{x_A}{C}}) \quad (3.12)$$

$$L - h = C(-e^{-\frac{x_B}{C}} + e^{-\frac{x_A}{C}}) \quad (3.13)$$

multiplicando las ecuaciones (3.12) y (3.13) tenemos:

$$L^2 - h^2 = C^2(e^{\frac{(x_B - x_A)}{C}} + e^{\frac{(x_B - x_A)}{C}} - 2) \quad (3.14)$$

del gráfico: $x_B - x_A = a$, reemplazando en (3.14):

$$L^2 - h^2 = C^2(e^{\frac{a}{C}} + e^{\frac{a}{C}} - 2)$$

$$L^2 - h^2 = C^2(e^{\frac{a}{2C}} - e^{-\frac{a}{2C}})^2$$

$$L^2 - h^2 = 2^2 C^2 \left(\frac{e^{\frac{a}{2C}} - e^{-\frac{a}{2C}}}{2} \right)^2 \quad (3.15)$$

sabemos que:

$$\sinh(p) = \left(\frac{e^p - e^{-p}}{2} \right)$$

reemplazando en (3.15):

$$L^2 - h^2 = 2^2 C^2 \left(\sinh\left(\frac{a}{2C}\right) \right)^2$$

$$L = \sqrt{\left[2C \sinh\left(\frac{a}{2C}\right) \right]^2 + h^2} \quad (3.16)$$

si $h=0$, entonces:

$$L = L' = 2C \sinh\left(\frac{a}{2C}\right)$$

que es la ecuación de la longitud de un cable a nivel, entonces la ecuación (3.16) se convierte en:

$$L = \sqrt{L'^2 + h^2} \quad (3.17)$$

3.6 Ecuación de la flecha para un cable a desnivel

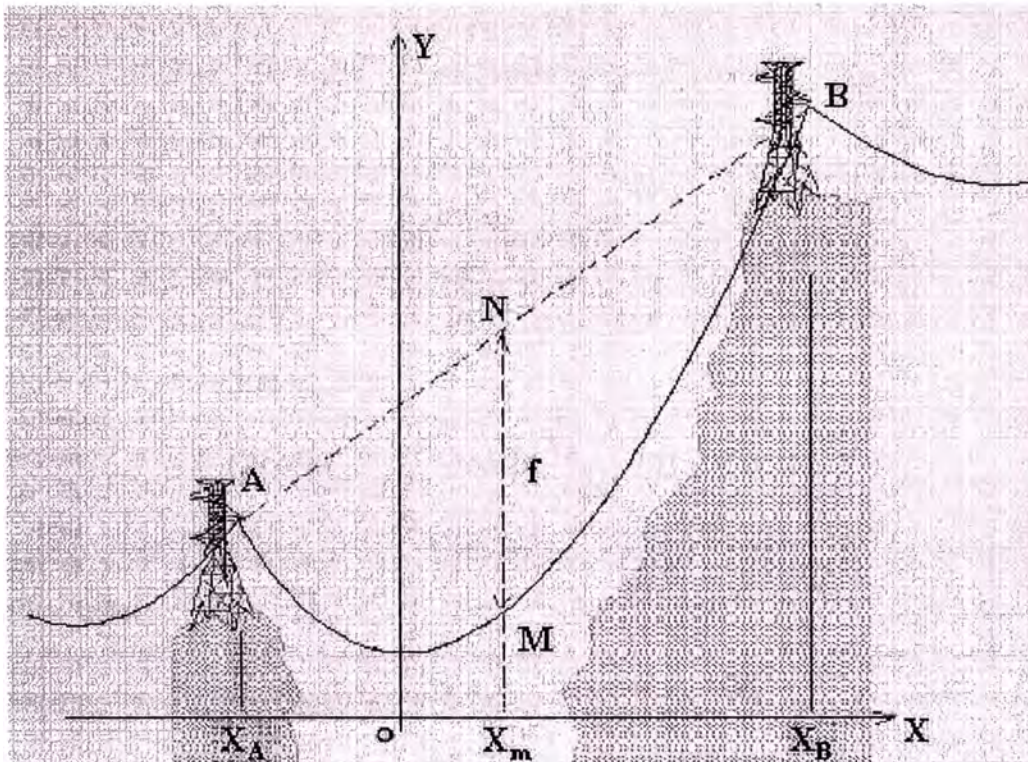


Fig. 3.3 Flecha para dos torres a desnivel

Sabemos que la ecuación de la flecha para un cable a nivel es:

$$f' = C \left(\cosh\left(\frac{x}{2C}\right) - 1 \right) \quad (3.18)$$

Observando el gráfico el valor de la flecha para un cable a desnivel es la diferencia de ordenadas entre los puntos N y M, entonces:

$$f = y_N - y_M \quad (3.19)$$

Como el punto P es el punto medio de AB:

$$y_N = \frac{1}{2}(y_A + y_B) \quad (3.20)$$

Reemplazando la ecuación (3.20) en la ecuación (3.19):

$$f = \frac{1}{2}(y_A + y_B) - y_M \quad (3.21)$$

reemplazando la ecuación de la catenaria en cada punto se obtiene:

$$f = \frac{1}{2}(C \cosh(\frac{x_A}{C}) + C \cosh(\frac{x_B}{C})) - C \cosh(\frac{x_M}{C}) \quad (3.22)$$

Sabemos que: $\cosh p + \cosh q = 2 \cosh(\frac{p+q}{2}) \cosh(\frac{p-q}{2})$

reemplazando en la ecuación (3.22):

$$f = \frac{C}{2} (2 \cosh(\frac{x_B + x_A}{2C}) \cosh(\frac{x_B - x_A}{2C})) - C \cosh(\frac{x_M}{C}) \quad (3.23)$$

pero sabemos que:

$$x_M = \frac{1}{2}(x_A + x_B) \quad y \quad a = x_B - x_A$$

reemplazando estos valores en la ecuación (3.23):

$$f = \frac{C}{2} (2 \cosh(\frac{x_M}{C}) \cosh(\frac{a}{2C})) - C \cosh(\frac{x_M}{C}) \quad (3.24)$$

simplificando nos queda:

$$f = C(\cosh(\frac{a}{2C}) - 1) \cosh(\frac{x_M}{C}) \quad (3.25)$$

si comparamos la ecuación (3.25) con la ecuación (3.18) nos damos cuenta que:

$$f = f' \cosh(\frac{x_M}{C}) \quad (3.26)$$

3.7 Ubicación cartesiana del punto medio y de los extremos del vano

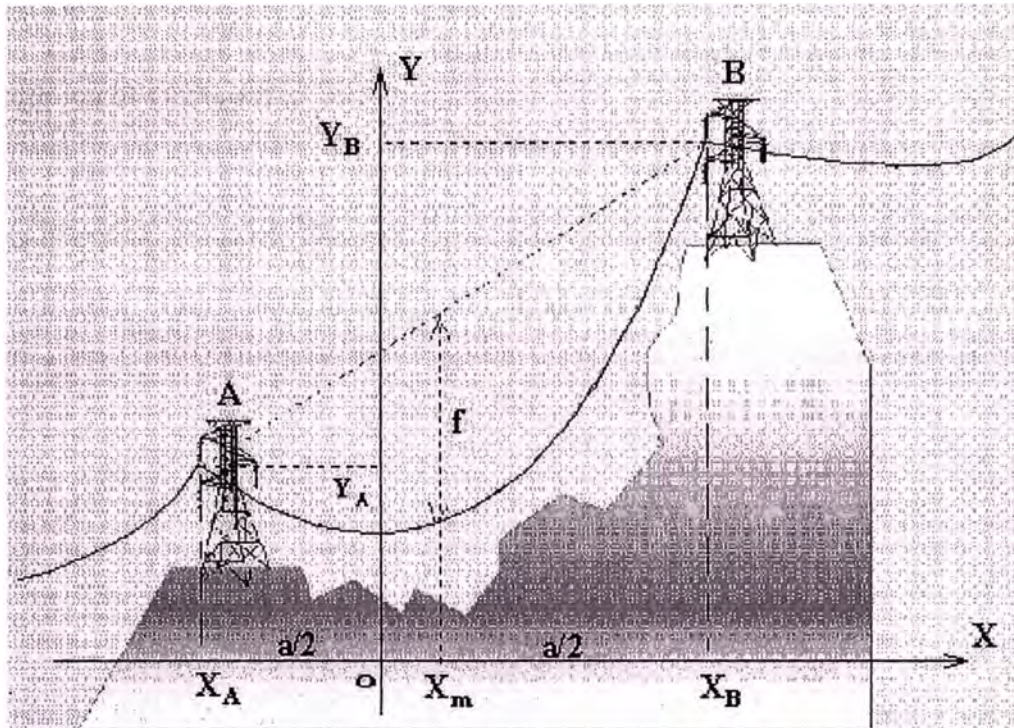


Fig. 3.4 Punto medio del vano

De la figura 3.4 tenemos:

$$x_A = x_m - \frac{a}{2} \quad (3.27)$$

$$x_B = x_m + \frac{a}{2} \quad (3.28)$$

Sabemos que la ecuación de desnivel es:

$$h = C \left[\cosh\left(\frac{x_B}{C}\right) - \cosh\left(\frac{x_A}{C}\right) \right]$$

reemplazando la ecuación (3.27) y (3.28), tenemos:

$$h = C \left[\cosh\left(\frac{x_m + \frac{a}{2}}{C}\right) - \cosh\left(\frac{x_m - \frac{a}{2}}{C}\right) \right] \quad (3.29)$$

sabemos que:

$$\cosh(p + q) = \cosh p \cosh q + \sinh p \sinh q$$

reemplazando en la ecuación (3.20):

$$h = C \left[\cosh\left(\frac{x_m}{C}\right) \cosh\left(\frac{a}{2C}\right) + \sinh\left(\frac{x_m}{C}\right) \sinh\left(\frac{a}{2C}\right) - \cosh\left(\frac{x_m}{C}\right) \cosh\left(\frac{a}{2C}\right) + \sinh\left(\frac{x_m}{C}\right) \sinh\left(\frac{a}{2C}\right) \right] \quad (3.30)$$

$$h = 2C \sinh\left(\frac{a}{2C}\right) \sinh\left(\frac{x_m}{C}\right) \quad (3.31)$$

sabemos que la longitud del cable a nivel es:

$$L' = 2C \sinh\left(\frac{a}{2C}\right)$$

entonces la ecuación (3.31) se convierte en:

$$h = L' \sinh\left(\frac{x_m}{C}\right) \quad (3.32)$$

despejando el valor de x_m :

$$x_m = C \sinh^{-1}\left(\frac{h}{L'}\right) \quad (3.33)$$

Al reemplazar el valor de x_m en las ecuaciones (3.27) y (3.28), obtenemos los valores de las abscisas en los puntos medios de las torres anterior (x_A) y posterior (x_B) al vano.

3.8 Parámetro de la catenaria en función del tiro máximo

A partir de la ecuación de la catenaria:

$$y = C \cosh\left(\frac{x}{C}\right)$$

que es igual a:

$$y = \frac{T_0}{w_c} \cosh\left(\frac{x}{C}\right) \rightarrow y w_c = T_0 \cosh\left(\frac{x}{C}\right) \rightarrow T_x = T_0 \cosh\left(\frac{x}{C}\right)$$

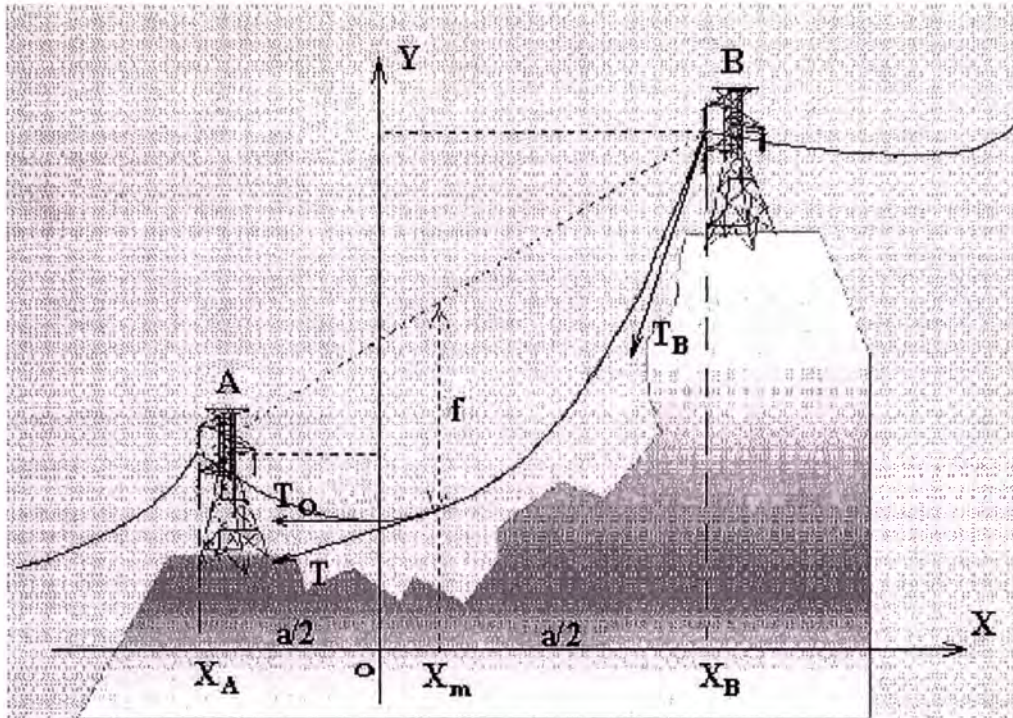


Fig. 3.5 Tiro máximo en torre B

el producto yw_c tiene las mismas unidades (Kg) que el segundo miembro. Son constantes: w_c , T_0 . Por tanto la ecuación es (digamos) Kg en función de x , es decir el Tiro en un punto cualquiera del cable. Finalmente entonces si cualquier ordenada de un punto del cable lo multiplicamos por w_c , resulta que es precisamente el Tiro (valor escalar) en ese mismo punto.

Por tanto resulta correcto que por ejemplo:

$$T_B = T_m + w_c \underbrace{\left(f + \frac{h}{2} \right)}_{\substack{\text{Ordenada} \\ \text{Tiro (kg)}}$$

Entonces el tiro máximo una vez instalado el cable se encuentra en el extremo superior entre dos torres a desnivel.

$$T_B = T_M + w_c \left(f + \frac{h}{2} \right) \quad (3.34)$$

de la ecuación (26) obtuvimos:

$$f = f' \cosh\left(\frac{x_M}{C}\right)$$

donde f' es la longitud del cable a nivel que mediante una aproximación en serie de Taylor se convierte en:

$$f' = \frac{a^2}{8C} \quad (3.35)$$

reemplazando la ecuación (3.35) en la ecuación (3.26), tenemos:

$$f = \frac{a^2}{8C} \cosh\left(\frac{x_M}{C}\right) \quad (3.36)$$

De la ecuación (3.33) obtuvimos:

$$x_M = C \operatorname{senh}^{-1}\left(\frac{h}{L'}\right)$$

y tomando la aproximación $L' = a$ entonces tenemos:

$$x_M = C \operatorname{senh}^{-1}\left(\frac{h}{a}\right) \quad (3.37)$$

reemplazando la ecuación (3.37) en la ecuación (3.36), tenemos:

$$f = \frac{a^2}{8C} \cosh\left(\operatorname{senh}^{-1}\left(\frac{h}{a}\right)\right) \quad (3.38)$$

De las relaciones de funciones hiperbólicas:

$$\cosh(\operatorname{senh}^{-1}(p)) = \sqrt{1 + p^2}$$

aplicando a la ecuación (3.38):

$$f = \frac{a^2}{8C} \frac{\sqrt{a^2 + h^2}}{a} \quad (3.39)$$

del grafico observamos que: $b = \sqrt{a^2 + h^2}$

reemplazando en la ecuación (3.39):

$$f = \frac{ab}{8C} \quad (3.40)$$

que es una ecuación aproximada para el cálculo de la flecha para cables en desnivel.

Si reemplazamos la ecuación (3.40) en la ecuación (3.34), tenemos:

$$T_B = T_M + \varpi_c \left(\frac{ab}{8C} + \frac{h}{2} \right) \quad (3.41)$$

Como se demostró anteriormente:

$$T_M = T_0 \cosh \left(\frac{x_M}{C} \right) \quad (3.42)$$

que tomando dos términos de la expansión de Taylor para el coseno hiperbólico, la ecuación (3.42) se convierte en:

$$T_M = T_0 \left(1 + \frac{\left(\frac{x_M}{C} \right)^2}{2} \right) \quad (3.43)$$

Reemplazando la ecuación (3.37) en la ecuación (3.43):

$$T_M = T_0 \left(1 + \frac{1}{2} \left(\operatorname{senh}^{-1} \left(\frac{h}{a} \right) \right)^2 \right) \quad (3.44)$$

como conocemos el valor del desnivel “h” y el vano “a” asumimos una constante k:

$$k = 1 + \frac{1}{2} \left(\operatorname{senh}^{-1} \left(\frac{h}{a} \right) \right)^2 \quad (3.45)$$

entonces:

$$T_M = T_0 K \quad (3.46)$$

reemplazando la ecuación (3.46) en la ecuación (3.41):

$$T_B = T_0 K + \varpi_c \left(\frac{ab}{8C} + \frac{h}{2} \right) \quad (3.47)$$

dividiendo la ecuación (3.47) por ϖ_c :

$$\frac{T_B}{\varpi_c} = \frac{T_0 K}{\varpi_c} + \left(\frac{ab}{8C} + \frac{h}{2} \right) \quad (3.48)$$

reemplazando el parámetro de la catenaria “C”:

$$\frac{T_B}{\varpi_c} = CK + \left(\frac{ab}{8C} + \frac{h}{2} \right) \quad (3.49)$$

La ecuación (3.49) es una ecuación de segundo grado en función de “C”, que al despejar la raíz positiva se obtiene

$$C = \frac{1}{2} \left[\frac{1}{K} \left(\frac{T_B}{\varpi_c} - \frac{h}{2} \right) + \sqrt{\left(\frac{1}{K} \left(\frac{T_B}{\varpi_c} - \frac{h}{2} \right) \right)^2 - \frac{ab}{2K}} \right] \quad (3.50)$$

que es la ecuación para obtener el parámetro de la catenaria de un cable a desnivel en función del tiro máximo “ T_B ”.

CAPITULO IV

PLANTEAMIENTO DEL MÉTODO DE SOLUCIÓN DEL CÁLCULO DE CORRECCIÓN DE ALINEAMIENTO VERTICAL DE CADENAS DE AISLADORES

4.1 Definición de corrección de alineamiento vertical de cadena de aisladores

El cálculo de los valores de flecha y corrimiento de cadena esta basado en el hecho que para las series de vanos entre terminales, la longitud total de los conductores en las roldanas debe ser igual al total de longitud en las grapas, usando el parámetro de la catenaria del vano principal. Entonces, los parámetros de la catenaria de cada vano individual están determinados por el conductor en las roldanas.

Donde los parámetros de la catenaria del vano individual con el conductor en las roldanas, difieren del parámetro de la catenaria del vano principal, resultarán diferentes flechas para el conductor del vano individual para las dos condiciones. Esta diferencia en las flechas de los conductores es la corrección de flecha. Así mismo, donde los parámetros de la catenaria del vano individual con el conductor en las roldanas, difieren del parámetro de la catenaria del vano principal, diferentes longitudes de arco del conductor del vano individual resultaran para las dos condiciones. Esta diferencia en las longitudes de arco del conductor, ajustadas para contabilizar los cambios que ocurren cuando la tensión se dirige de la tensión del flechado a la tensión de la grapa, es la corrección de alineamiento vertical de la cadena de aisladores, también conocida como corrimiento de cadena y en el idioma inglés como “clipping offset”.

4.2 Procedimiento de cálculo

- A) Se selecciona un vano que generalmente es el más accesible o el más próximo a la ultima torre de anclaje y se le denomina “vano de control”, porque es en este vano donde vamos a medir el tiro de templado y la flecha.

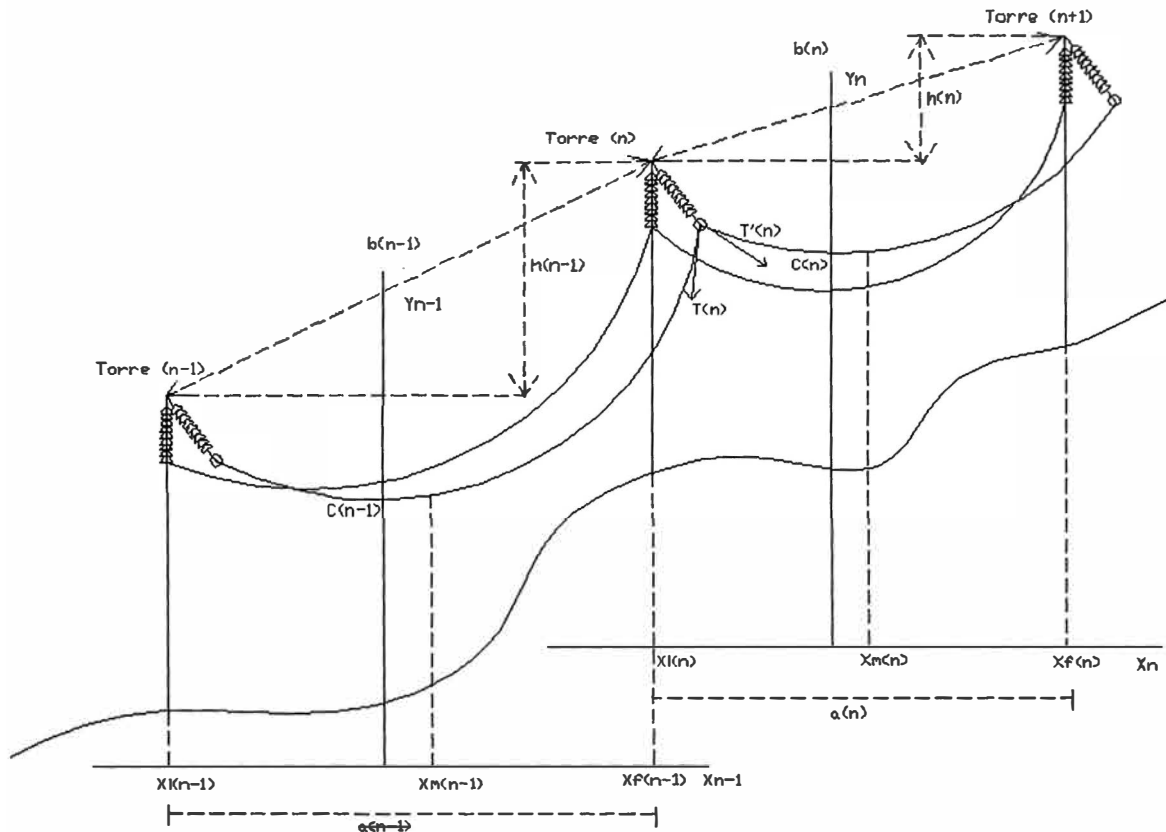


Fig. 4.1 Diagrama de dos vanos adyacentes

B) Para los vanos que se encuentran a la izquierda del vano de control se procede de la siguiente manera:

1º Se calcula el parámetro de la catenaria (C) del vano de control mediante:

$$C_{(n)} = \frac{T_{0(n)}}{\omega} \quad (4.1)$$

2º Se calcula la longitud del cable en el vano de control como si estuviera a nivel mediante:

$$L'_{(n)} = 2C_{(n)} \operatorname{senh}\left(\frac{a}{2C_{(n)}}\right) \quad (4.2)$$

3º Se calcula el valor de la abscisa en el punto medio del vano:

$$x_{m(n)} = C_{(n)} \operatorname{senh}^{-1}\left(\frac{h_{(n)}}{L'_{(n)}}\right) \quad (4.3)$$

4º Se calcula el valor de la abscisa en la torre anterior:

$$x_{i(n)} = x_{m(n)} - \frac{a_{(n)}}{2} \quad (4.4)$$

5° Se calcula el tiro en la torre anterior:

$$T_{(n-1)} = T'_{(n)} = T_{o(n)} \cosh\left(\frac{x_{i(n)}}{C_{(n)}}\right) \quad (4.5)$$

6° Se calcula el parámetro de la catenaria del vano anterior (n-1):

$$C_{(n-1)} = \frac{1}{2} \left[\frac{1}{K_{(n-1)}} \left(\frac{T_{(n-1)}}{\varpi_c} - \frac{h_{(n-1)}}{2} \right) + \sqrt{\left(\frac{1}{K_{(n-1)}} \left(\frac{T_{(n-1)}}{\varpi_c} - \frac{h_{(n-1)}}{2} \right) \right)^2 - \frac{a_{(n-1)} b_{(n-1)}}{2K_{(n-1)}}} \right] \quad (4.6)$$

$$\text{donde, } b_{(n-1)} = \sqrt{a_{(n-1)}^2 + h_{(n-1)}^2}$$

$$K_{(n-1)} = 1 + \frac{1}{2} \left[\operatorname{senh}^{-1} \left(\frac{h_{(n-1)}}{a_{(n-1)}} \right) \right]^2$$

7° Se calcula Tiro Horizontal en al vano anterior (n-1):

$$T_{0(n-1)} = w_c C_{n-1} \quad (4.7)$$

8° Se procede de la misma forma hasta llegar al último vano de la izquierda.

C) Para los vanos que se encuentran a la derecha del vano de control se procede de la siguiente manera:

1° Se calcula en el vano de control el valor de la abscisa en la torre posterior:

$$x_{f(n)} = x_{m(n)} + \frac{a_{(n)}}{2} \quad (4.8)$$

2° Se calcula el tiro en la torre posterior:

$$T'_{(n+1)} = T_{(n)} = T_{o(n)} \cosh\left(\frac{x_{f(n)}}{C_{(n)}}\right) \quad (4.9)$$

3° Se calcula el parámetro de la catenaria del vano posterior:

$$C_{(n+1)} = \frac{1}{2} \left[\frac{1}{K_{(n+1)}} \left(\frac{T_{(n+1)}}{\omega_c} + \frac{h_{(n+1)}}{2} \right) + \sqrt{\left(\frac{1}{K_{(n+1)}} \left(\frac{T_{(n+1)}}{\omega_c} + \frac{h_{(n+1)}}{2} \right) \right)^2 - \frac{a_{(n+1)} b_{(n+1)}}{2K_{(n+1)}}} \right] \quad (4.10)$$

donde, $b_{(n+1)} = \sqrt{a_{(n+1)}^2 + h_{(n+1)}^2}$

$$K_{(n+1)} = 1 + \frac{1}{2} \left[\sinh^{-1} \left(\frac{h_{(n+1)}}{a_{(n+1)}} \right) \right]^2$$

4° Se calcula Tiro Horizontal en al vano anterior (n-1):

$$T_{0(n+1)} = w_c C_{n+1} \quad (4.11)$$

5° Se procede de la misma forma hasta llegar al ultimo vano de la derecha.

D) Calcular el peso de conductor que soporta la cadena de aisladores

1° En el vano $a_{(n-1)}$ (ejes X_{n-1} - Y_{n-1}) de la figura 4.1, la longitud entre el vértice de la catenaria y el punto de suspensión es L_{xn} :

$$L_{xn} = C_{(n-1)} \sinh \frac{x_{f(n-1)}}{C_{(n-1)}} \quad (4.12)$$

2° En el vano a_n (ejes X_n - Y_n) de la figura 4.1, la longitud entre el vértice de la catenaria y el punto de suspensión es $L_{x'n}$:

$$L_{x'n} = -C_{(n)} \sinh \frac{x_{i(n)}}{C_{(n)}} \quad (4.13)$$

3° La longitud total del conductor “ L_n ” que soporta la cadena (n), cuyo peso total influye en el ángulo de la cadena es.

$$L_n = L_{xn} + L_{x'n} \quad (4.14)$$

4° En consecuencia, el peso total del conductor que soporta la cadena (n) que resulta de multiplicar la ecuación 4.14 por w_c , es:

$$w_c L_n = w_c (L_{xn} + L_{x'n}) \quad (4.15)$$

E) Determinar el valor del ángulo δ que hace la cadena de aisladores con respecto a la vertical, por efecto del Tiro de Tendido.

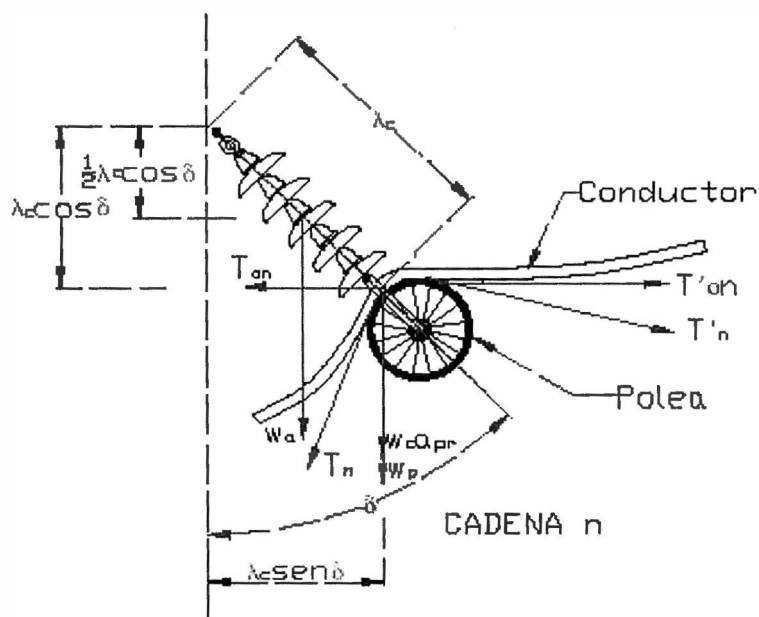


Fig. 4.2 Diagrama de fuerzas sobre la roldana

Donde:

T'_n = Tiro (Kg) hacia la derecha originado por el winche y por el peso propio del conductor.

T_n = Tiro (Kg) hacia la izquierda.

T'_{on} = Tiro (Kg) horizontal correspondiente a T'_n

T_{on} = Tiro (Kg) horizontal correspondiente a T_n

$w_c a_{pr}$ = Tiro (Kg) hacia abajo debido al peso del conductor.

w_c = peso (Kg/m) unitario del conductor.

w_a = peso (Kg) de la cadena de aisladores.

W_p = peso (Kg) de la polea o roldana.

a_{pr} = vano peso (m) que soporta la polea.

λ_c = longitud (m) de la cadena de aisladores.

δ = ángulo ($^\circ$ Sexagesimales) que hace la cadena respecto a la vertical.

Respecto al punto de suspensión de la cadena, tomemos suma de momentos, lo que resulta entonces:

$$w_a \frac{\lambda_c}{2} \text{sen} \delta + (w_c L_p + w_p) \text{sen} \delta = (T'_{on} - T_{on}) \lambda_c \cos \delta \quad (4.16)$$

Eliminando λ_c y agrupando convenientemente resulta entonces:

$$\left(\frac{w_a}{2} + w_c a_{pr} + w_p \right) \text{sen} \delta = (T'_{on} - T_{on}) \cos \delta \quad (4.17)$$

Por tanto:

$$\text{tg}(\delta) = \left(\frac{T'_{on} - T_{on}}{\frac{w_a}{2} + w_c a_{pr} + w_p} \right) \quad (4.18)$$

Finalmente

$$\delta = \text{arctg} \left(\frac{T'_{on} - T_{on}}{\frac{w_a}{2} + w_c a_{pr} + w_p} \right) \quad (4.19)$$

Los valores de T'_{on} y T_{on} , los obtenemos de las ecuaciones 4.7 y 4.11, según corresponda.

F) Calcular los desplazamientos horizontales (Δa) y verticales (Δh) de las cadenas:

Los desplazamientos horizontales de cada una de las cadenas son (Fig 4.3):

$$e_{n-1} = \lambda_c \text{sen} \delta_{n-1} \quad (4.20)$$

$$e_n = \lambda_c \text{sen} \delta_n \quad (4.21)$$

El desplazamiento horizontal neto será:

$$\Delta a = e_n - e_{n-1} = \lambda_c (\text{sen} \delta_n - \text{sen} \delta_{n-1}) \quad (4.22)$$

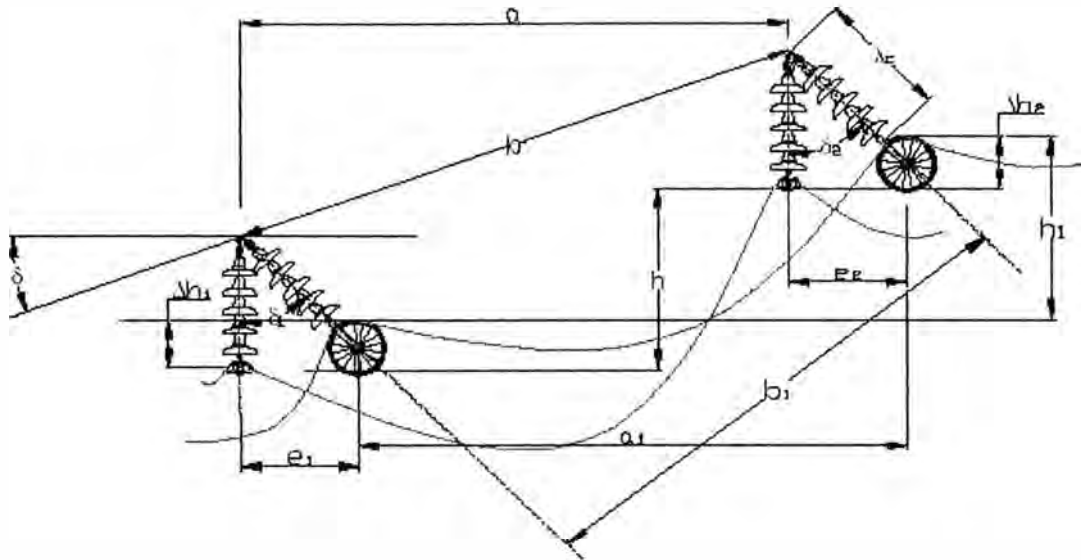


Fig. 4.3 Diagrama de desplazamientos en un vano con el cable en roldanas

Donde:

a_n = vano (m) horizontal, con cadenas engrapadas.

a_{1n} = vano (m) horizontal, con cadenas en poleas.

b_n = vano real (m) entre los puntos de suspensión del conductor con cadenas engrapadas.

b_{1n} = vano real (m) entre los puntos de suspensión del conductor con cadenas en poleas.

h_n = desnivel (m) con conductor engrapado.

h_{1n} = desnivel (m) con conductor engrapado en poleas.

e_{n-1}, e_n = desplazamientos horizontales (m) de los puntos de suspensión del conductor con poleas.

$\Delta h_{n-1}, \Delta h_n$ = desplazamientos verticales (m) de los puntos de suspensión del conductor con poleas.

δ_{n-1}, δ_n = ángulos de desviación ($^\circ$ sexagesimales) de las cadenas con poleas.

λ_c = longitud (m) de la cadena de aisladores.

por tanto el nuevo vano horizontal con el conductor en poleas es:

$$a'_n = a + \Delta a = a + \lambda_c (\text{sen} \delta_n - \text{sen} \delta_{n-1}) \quad (4.23)$$

Respecto a los desplazamientos verticales con cadenas en poleas.

$$\Delta h_{n-1} = \lambda_c - \lambda_c \cos \delta_{n-1} = \lambda_c (1 - \cos \delta_{n-1}) \quad (4.24)$$

$$\Delta h_n = \lambda_C - \lambda_C \cos \delta_n = \lambda_C (1 - \cos \delta_n) \quad (4.25)$$

El desplazamiento vertical neto será:

$$\Delta h = \Delta h_n - \Delta h_{n-1} = \lambda_C (\cos \delta_{n-1} - \cos \delta_n) \quad (4.26)$$

por tanto el nuevo desnivel con el conductor en poleas es:

$$h'_n = h + \lambda_C (\cos \delta_{n-1} - \cos \delta_n) \quad (4.27)$$

El valor del ángulo δ_{n-1} y δ_n lo obtenemos de la ecuación 4.19

G) Calcular la longitud de conductor en poleas en cada vano

$$L'_{n \text{ poleas}} = 2C_n \operatorname{senh} \left(\frac{a'_n}{2C_n} \right) \quad (4.28)$$

$$L_{n \text{ poleas}} = \sqrt{L'^2_{n \text{ poleas}} + h'^2_n} \quad (4.29)$$

H) Calcular la longitud de conductor en grapas en cada vano

$$L'_{n \text{ grapa}} = 2C_n \operatorname{senh} \left(\frac{a_n}{2C_n} \right) = 2C \operatorname{senh} \left(\frac{a_n}{2C} \right) \quad (4.30)$$

siendo C constante a lo largo de todo el Tramo:

$$L_{n \text{ grapa}} = \sqrt{L'^2_{n \text{ grapa}} + h_n^2} \quad (4.31)$$

I) Calcular la corrección de alineamiento vertical de la cadena de aisladores (ΔL_n):

$$\Delta L_n = L_{\text{poleas}} - L_{\text{grapas}} \quad (4.32)$$

CAPITULO V DESARROLLO DEL PROGRAMA APLICATIVO

5.1 Software utilizado

El presente sistema ha sido realizado utilizando el lenguaje de programación web PHP y la base de datos MYSQL, los cuales pertenecen al software “open source”, es decir que no se requieren licencias para su uso. Ver detalles de este software en el anexo A.

5.2 Metodología utilizada

Siguiendo la tendencia actual, en cuanto a desarrollo de sistemas, para el desarrollo del siguiente sistema se ha utilizado la tecnología web y la programación orientada a objetos. Se ha aplicado también la metodología de vista controlada, que permite separar la parte del diseño (página web), de la parte lógica (programación).

Se han desarrollado clases que representan a las principales entidades involucradas en el tendido de una línea de transmisión, como son:

- Línea
- Tramo
- Vano

Así como las clases auxiliares de usuarios, utilitarios y conexión a la base de datos:

- GrupoUsuario
- Usuario
- Conexión
- Util

Para el detalle de las clases, ver anexo B.

Se ha desarrollado un diseño básico (prototipo), el cuál puede cambiar en cualquier momento a solicitud del cliente sin necesidad de hacer cambios en la programación, solamente se hacen los cambios en el lado de presentación (página web). Esto es una ventaja de utilizar la metodología orientada a objetos.

5.3 Manual de Usuario

- El primer paso es ingresar el usuario y clave autorizados:

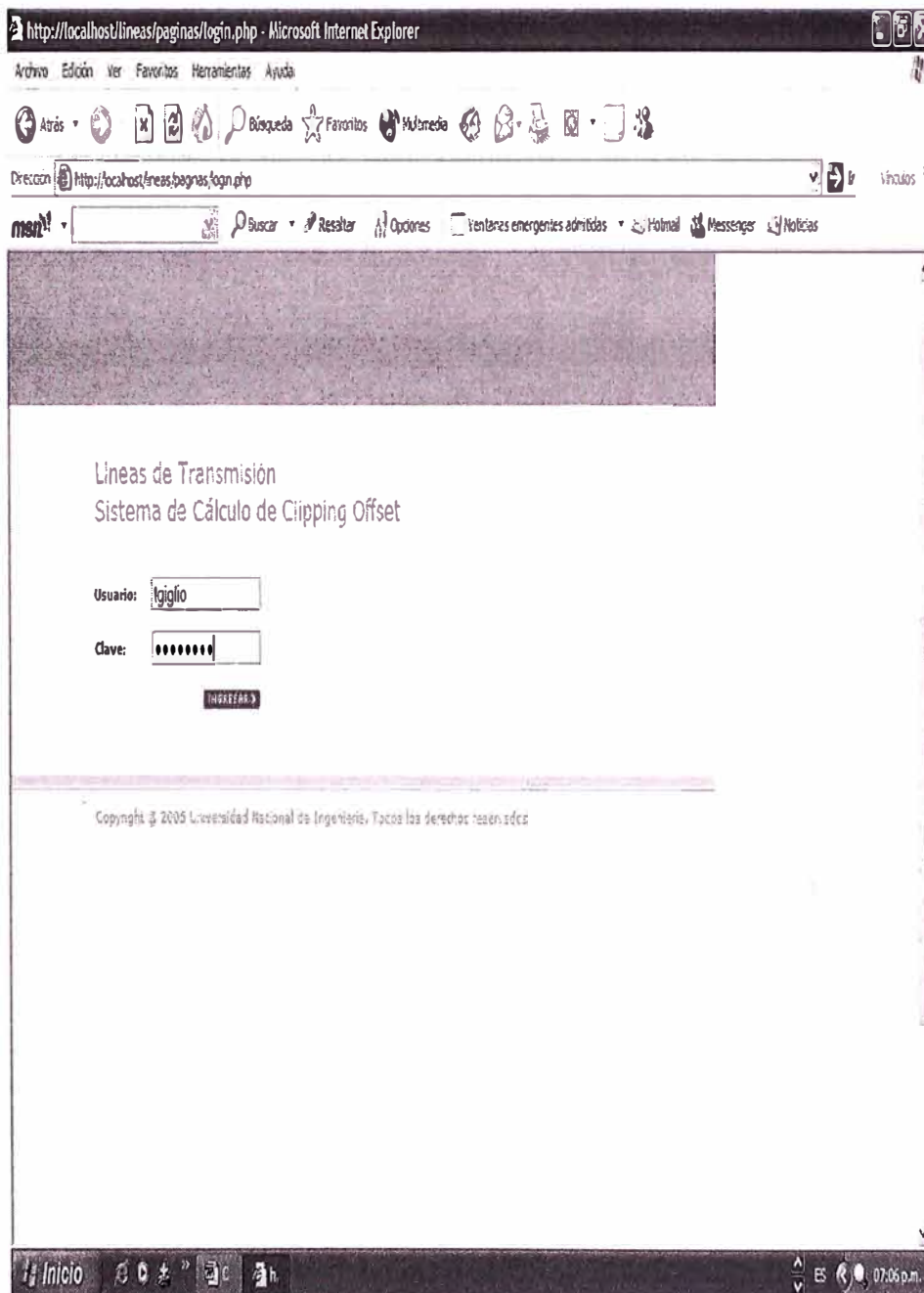


Fig. 5.1 Pantalla de inicio de sesión

- Seguidamente se mostrará la pantalla de bienvenida, donde se indicará los pasos a seguir:

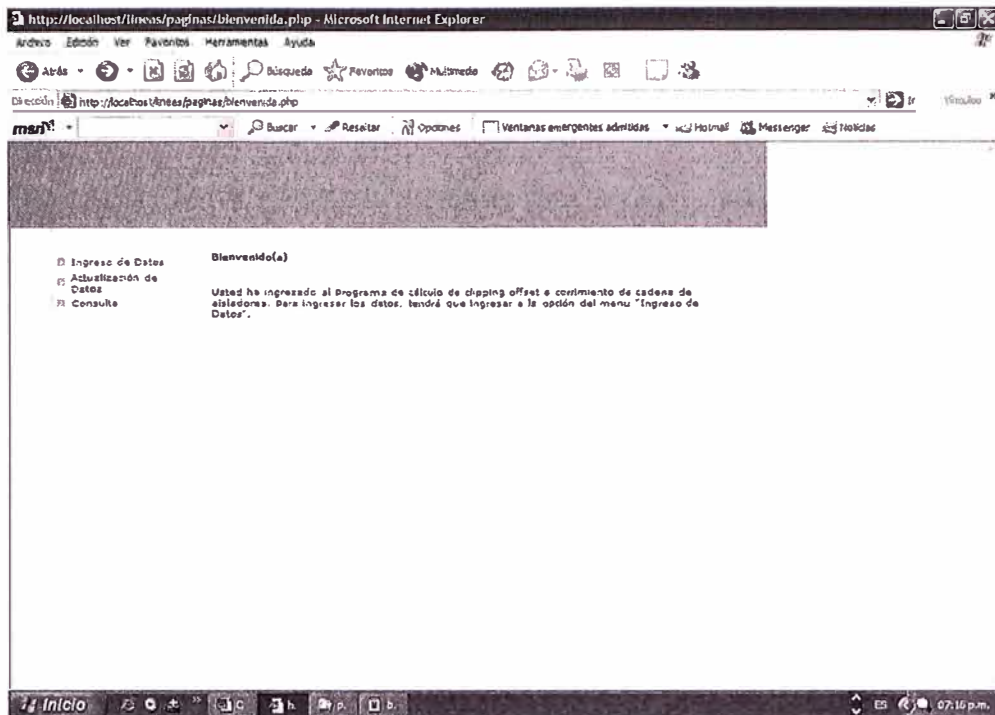


Fig. 5.2 Pantalla de bienvenida

- Para el ingreso de datos se debe entrar a la opción de menú “Ingreso de Datos”, donde inicialmente se mostrarán los datos de la línea de transmisión:

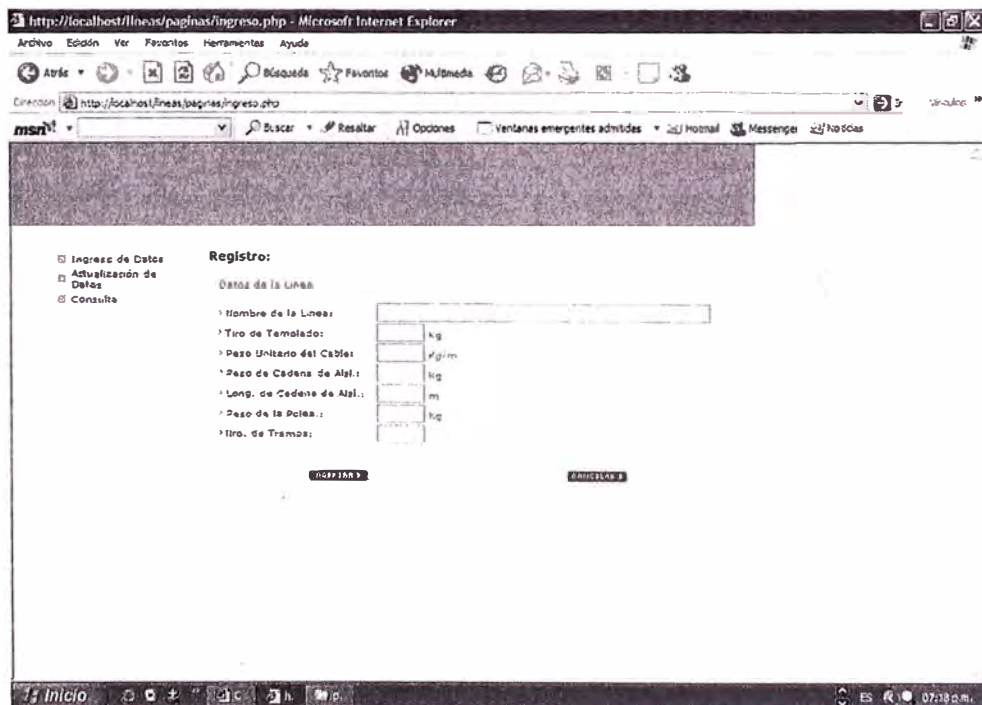


Fig. 5.3 Pantalla de ingreso de datos de la línea

- Se debe ingresar los datos de los tramos:

Registro:

Datos de la Línea

Nombre de la Línea: Línea de Prueba

Tipo de Templado: 2400 Kg

Peso Unitario del Cable: 1.522 Kg/m

Peso de Cadena de Aisl.: 75 Kg

Long. de Cadena de Aisl.: 2.4 m

Peso de la Polea: 25 Kg

Nro. de Tramos: 5

Datos del Tramo

Nombre del Tramo: Tramo 1

Nro. de Vanos: 5

Vano de Control: 5

ACEPTAR CANCELAR

Fig. 5.6 Pantalla de grabación de los datos del tramo

- Luego que se grabaron los datos del tramo, aparecen las opciones para ingresar los datos de los vanos:

Registro:

Datos de la Línea

Nombre de la Línea: Línea de Prueba

Tipo de Templado: 2400 Kg

Peso Unitario del Cable: 1.522 Kg/m

Peso de Cadena de Aisl.: 75 Kg

Long. de Cadena de Aisl.: 2.4 m

Peso de la Polea: 25 Kg

Nro. de Tramos: 5

Datos del Tramo

Nombre del Tramo: Tramo 1

Nro. de Vanos: 5

Vano de Control: 5

Datos de los Vanos

Nro. Vano	Distancia (m)	Altura (m)	Delta "h" (m)
1			
2			
3			
4			
5			

ACEPTAR CANCELAR

Fig. 5.7 Pantalla de ingreso de datos de los vanos

- Se deben ingresar los datos de los vanos:

Registro:

Datos de la Línea

Nombre de la Línea: Linea de Prueba

Tiro de Templado: 2400 Kg

Peso Unitario del Cable: 1.522 Kg/m

Peso de Cadena de Aisl.: 75 Kg

Long. de Cadena de Aisl.: 2.4 m

Peso de la Polea.: 25 Kg

Nro. de Tramos: 5

Datos del Tramo

Nombre del Tramo: Tramo 1

Nro. de Vanos: 5

Vano de Control: 5

Datos de los Vanos

NºVano	Distancia (m)	Altura (m)	Delta "L" (m)
1	369.1	-7.5	
2	413	9.1	
3	672.1	73.4	
4	219.7	-15.5	
5	632.1	-43.2	

Fig. 5.8 Pantalla de grabación de los datos de los vanos

- Una vez que todos los datos se han ingresado, se presiona el botón “Aceptar” para realizar el cálculo del clipping offset de cada vano:

Registro:

Datos de la Línea

Nombre de la Línea: Linea de Prueba

Tiro de Templado: 2400 Kg

Peso Unitario del Cable: 1.522 Kg/m

Peso de Cadena de Aisl.: 75 Kg

Long. de Cadena de Aisl.: 2.4 m

Peso de la Polea.: 25 Kg

Nro. de Tramos: 5

Datos del Tramo

Nombre del Tramo: Tramo 1

Nro. de Vanos: 5

Vano de Control: 5

Datos de los Vanos

NºVano	Distancia (m)	Altura (m)	Delta "L" (m)
1	369.1	-7.5	0.00
2	413	9.1	0.00
3	672.1	73.4	0.00
4	219.7	-15.5	0.00
5	632.1	-43.2	0.00

Fig. 5.9 Pantalla que muestra el cálculo final

En la columna Delta "L" se ha calculado el valor que hay que correr la grapa sobre el conductor, según sea el signo negativo o positivo, respectivamente a la izquierda y derecha de la marca vertical de la plomada que se realizó con el conductor en roldanas. Esta afirmación es válida si se considera que el primer vano (Nº Vano =1) esta a la izquierda y los siguientes vanos (2,3,4,5) a la derecha de él.

Al valor de Delta "L", se le conoce como corrimiento de cadena de aisladores o clipping offset.

CAPITULO VI

APLICACIÓN A UN CASO PRÁCTICO DE LA REALIDAD NACIONAL

6.1 Datos de la línea de transmisión

Para la aplicación del caso práctico se evaluará la línea PARAMONGA-HUARMEY de 66 KV. Los datos de la línea son los siguientes:

- Tiro de templado = 641 Kg
- Peso unitario del cable = 0.19 Kg/m
- Peso de la cadena de aisladores = 45 Kg
- Longitud de la cadena de aisladores = 1.10 m
- Peso de la polea = 25 Kg

6.2 Datos del tramo

El tramo con el que vamos a trabajar esta comprendido entre la torre 28 a la 35 de la línea de transmisión Paramonga - Huarmey.

VANO	LONGITUD	COTA ANT	COTA POS	Delta H
1	200.41	91	83.02	-7.98
2	612.66	83.02	69.45	-13.57
3	492	69.45	31.7	-37.75
4	250	31.7	33.4	1.7
5	221	33.4	34.25	0.85
6	220.84	34.25	34.12	-0.13
7	190	34.12	35.18	1.06
8	201	35.18	36.09	0.91
9	210	36.09	37.2	1.11
10	218	37.2	33.61	-3.59
11	194.03	33.61	34.17	0.56
12	279.49	34.17	50.65	16.48

6.3 Cálculo del corrimiento de cadena utilizando el programa aplicativo

Registro:

Datos de la Línea

> Nombre de la Línea:

> Tiro de Templado: Kg

> Peso Unitario del Cable: Kg/m

> Peso de Cadena de Aisl.: Kg

> Long. de Cadena de Aisl.: m

> Peso de la Poles.: Kg

> Hro. de Tramos:

Datos del Tramo

> Nombre del Tramo:

> Hro. de Vanos:

> Vano de Control:

Datos de los Vanos

NºVano	Distancia (m)	Altura (m)	Delta "L" (m)
1	200.41	-7.98	0.06
2	612.65	-13.57	0.05
3	492	-37.75	0.02
4	250	1.7	0.01
5	221	0.85	0.01
6	220.84	-0.13	0.02
7	190	1.05	0.01
8	201	0.91	0.01
9	210	1.11	0.02
10	218	-3.59	-0.01
11	194.03	0.55	0.01
12	279.49	16.48	0.01

Fig. 6.1 Pantalla que muestra los resultados obtenidos

De los resultados obtenidos se deduce que:

- En la segunda torre se debe correr la grapa una distancia de 6 cm. a la izquierda sobre el conductor.
- En la tercera torre se debe correr la grapa una distancia de 6 cm. a la derecha sobre el conductor.
- En la cuarta torre se debe correr la grapa una distancia de 3 cm. a la izquierda sobre el conductor.

- En las torres 5 y 9 se debe correr la grapa una distancia de 1 cm. a la izquierda sobre el conductor.
- En las torres 11 y 12 se debe correr la grapa una distancia de 1 cm. a la derecha sobre el conductor.
- En las demás torres no se debe realizar ninguna corrección.

Si se realizan todas las correcciones anteriores las cadenas de aisladores deben quedar perfectamente verticales luego de terminado el engrapado del conductor.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. En el presente informe se ha demostrado la validez del método de cálculo de corrección vertical de cadena de aisladores propuesto, y además se ha desarrollado un programa aplicativo para que sea utilizado en el tendido de líneas de transmisión.
2. El método planteado debe ser utilizado cuando existe diferencia de cotas o alturas entre las torres de transmisión correspondientes a un tramo de la línea.
3. En el mercado existe software que realiza el cálculo integral de una línea de transmisión, pero se necesita pagar una licencia para su uso. En este informe se ha desarrollado un programa aplicativo que resuelve un problema de ingeniería del tendido de una línea de transmisión, utilizando software libre de licencia y obteniendo resultados rápidos, consistentes y almacenados en una base de datos.
4. El uso de la tecnología web para el desarrollo del programa aplicativo hace que este pueda ser usado desde cualquier parte del mundo en que se cuente con una conexión a Internet (puede ser satelital), sin necesidad de que los programas ejecutables estén físicamente en el computador utilizado.

RECOMENDACIONES

1. Se espera que este programa aplicativo sea un punto de partida para seguir realizando aplicativos utilizando la tecnología web, para que así el conocimiento se propague más rápido a través de Internet.
2. Se recomienda utilizar los programas libres de licenciamiento y la metodología de programación orientada a objetos por ser una buena práctica de programación y el código puede ser reutilizable para otros aplicativos.

ANEXO A
SOFWARE Y BASE DE DATOS

A.1 Definición del lenguaje de programación PHP

El PHP (acrónimo de "PHP: Hypertext Preprocessor"), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. Traduciendo la definición del FAQ de [PHP.net](http://www.php.net): *"El PHP es un lenguaje de script incrustado dentro del HTML. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de paginas"*.

Con PHP se puede hacer procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. Un sitio con paginas dinámicas es el que permite interactuar con el visitante, de modo que cada usuario que visita la pagina vea la información modificada para requisitos articulares. Las aplicaciones dinámicas para el Web son frecuentes en los sitios comerciales (e-commerce), donde el contenido visualizado se genera de la información alcanzada en una base de datos o fuente externa.

Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Entre su soporte pueden mencionarse InterBase, mSQL, MySQL, Oracle, Informix, PosgreSQL, entre otras.

PHP también ofrece la integración con las varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa desde generar documentos en pdf hasta analizar código XML.

PHP ofrece una solución simple y universal para las paginaciones dinámicas del Web de fácil programación. Su diseño elegante lo hace perceptiblemente más fácil de mantener y realizar actualizaciones en el código comparables en otros lenguajes. Debido a su amplia distribución PHP esta perfectamente soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.

PHP es utilizado en aplicaciones Web-relacionadas por algunas de las organizaciones más prominentes tales como Mitsubishi, Redhat, Der Spiegel, MP3-Lycos, Ericsson y NASA.

A.2 Definición de la base de datos Mysql

MySQL es uno de los Sistemas Gestores de bases de Datos (SQL) más populares desarrolladas bajo la filosofía de código abierto.

La desarrolla y mantiene la empresa MySql AB pero puede utilizarse gratuitamente y su código fuente está disponible.

A.2.1 Características (versión 4.0 en adelante)

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad; aquellos elementos faltantes fueron llenados por la vía de las aplicaciones que la utilizan.

Poco a poco los elementos faltantes en MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

A.2.2 MySQL en cifras

- Según las cifras del fabricante, existirían más de seis millones de copias de MySQL funcionando en la actualidad, lo que supera la base instalada de cualquier otra herramienta de bases de datos.
- El tráfico del sitio web de MySQL AB supera en 2004 el tráfico del sitio de IBM.

A.2.3 ¿Qué licencia utilizar?

La licencia GPL de MySQL obliga a distribuir cualquier producto derivado (aplicación) bajo esa misma licencia. Si un desarrollador desea incorporar MySQL en su producto pero no desea distribuirlo bajo licencia GPL, puede adquirir la licencia comercial de MySQL que le permite hacer justamente eso.

ANEXO B
LÓGICA DE PROGRAMACIÓN (CLASES)

B.1 Definición de clases utilizadas en el desarrollo del programa aplicativo

B.1.1 Clase Linea

```
<?php
include_once("../clases/Conexion.cls.php");

class Linea
{
    var $decCodiLine;
    var $strNombLine;
    var $strIdenUsua;
    var $decTiroHori;
    var $decPesoCond;
    var $decPesoCadeAisl;
    var $decAltuCadeAisl;
    var $decPesoRold;
    var $decTempInst;
    var $decNumeTram;

    function getdecCodiLine()
    {
        return $this->decCodiLine;
    }

    function setdecCodiLine($newdecCodiLine)
    {
        $this->decCodiLine=$newdecCodiLine;
    }

    function getstrNombLine()
    {
        return $this->strNombLine;
    }

    function setstrNombLine($newstrNombLine)
    {
        $this->strNombLine=$newstrNombLine;
    }
}
```

```
}

function getstrIdenUsua()
{
    return $this->strIdenUsua;
}

function setstrIdenUsua($newstrIdenUsua)
{
    $this->strIdenUsua=$newstrIdenUsua;
}

function getdecTiroHori()
{
    return $this->decTiroHori;
}

function setdecTiroHori($newdecTiroHori)
{
    $this->decTiroHori=$newdecTiroHori;
}

function getdecPesoCond()
{
    return $this->decPesoCond;
}

function setdecPesoCond($newdecPesoCond)
{
    $this->decPesoCond=$newdecPesoCond;
}

function getdecPesoCadeAisl()
{
    return $this->decPesoCadeAisl;
}

function setdecPesoCadeAisl($newdecPesoCadeAisl)
{
    $this->decPesoCadeAisl=$newdecPesoCadeAisl;
}
```

```
}

function getdecAltuCadeAisl()
{
    return $this->decAltuCadeAisl;
}

function setdecAltuCadeAisl($newdecAltuCadeAisl)
{
    $this->decAltuCadeAisl=$newdecAltuCadeAisl;
}

function getdecPesoRold()
{
    return $this->decPesoRold;
}

function setdecPesoRold($newdecPesoRold)
{
    $this->decPesoRold=$newdecPesoRold;
}

function getdecTempInst()
{
    return $this->decTempInst;
}

function setdecTempInst($newdecTempInst)
{
    $this->decTempInst=$newdecTempInst;
}

function getdecNumeTram()
{
    return $this->decNumeTram;
}

function setdecNumeTram($newdecNumeTram)
{
    $this->decNumeTram=$newdecNumeTram;
}
```

```

}

// Inserta todos los campos
// Devuelve un valor lógico
function agregar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia= "insert into linea      (TCODILINE
                ,TNOMBLINE
                ,TIDENUSUA
                ,TTIROHORI
                ,TPESOCOND
                ,TPESOCADAEISL
                ,TALTUCADAEISL
                ,TPESOROLD
                ,TTEMPINST
                ,TNUMETRAM
                )
                values      (".( $this-
>getdecCodiLine()==null?"null":$this->getdecCodiLine())."
                ,".".$this->getstrNombLine()."'
                ,".".$this->getstrIdenUsua()."'
                ,".($this-
>getdecTiroHori()==null?"null":$this->getdecTiroHori())."
                ,".($this-
>getdecPesoCond()==null?"null":$this->getdecPesoCond())."
                ,".($this-
>getdecPesoCadeAisl()==null?"null":$this->getdecPesoCadeAisl())."
                ,".($this-
>getdecAltuCadeAisl()==null?"null":$this->getdecAltuCadeAisl())."
                ,".($this-
>getdecPesoRold()==null?"null":$this->getdecPesoRold())."
                ,".($this-
>getdecTempInst()==null?"null":$this->getdecTempInst())."
                ,".($this-
>getdecNumeTram()==null?"null":$this->getdecNumeTram())."
                )";

    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);

```

```

        $objConexion->desconectar($link);
        return $bolResp;
    }

    // Actualiza sólo los campos que han sido seteados
    // Devuelve un valor lógico
    function actualizar()
    {
        $objConexion = new Conexion;
        $link=$objConexion->conectar();
        $sentencia="update linea set ";
        if ($this->getstrNombLine()!=null)
        $sentencia=$sentencia."TNOMBLINE='". $this->getstrNombLine()."', ";
        if ($this->getstrIdenUsua()!=null)
        $sentencia=$sentencia."TIDENUSUA='". $this->getstrIdenUsua()."', ";
        if ($this->getdecTiroHori()!=null)
        $sentencia=$sentencia."TTIROHORI=".$this->getdecTiroHori().", ";
        if ($this->getdecPesoCond()!=null)
        $sentencia=$sentencia."TPESOCOND=".$this->getdecPesoCond().", ";
        if ($this->getdecPesoCadeAisl()!=null)
        $sentencia=$sentencia."TPESOCADEAISL=".$this->getdecPesoCadeAisl().", ";
        if ($this->getdecAltuCadeAisl()!=null)
        $sentencia=$sentencia."TALTUCADEAISL=".$this->getdecAltuCadeAisl().", ";
        if ($this->getdecPesoRold()!=null)
        $sentencia=$sentencia."TPESOROLD=".$this->getdecPesoRold().", ";
        if ($this->getdecTempInst()!=null)
        $sentencia=$sentencia."TTEMPINST=".$this->getdecTempInst().", ";
        if ($this->getdecNumeTram()!=null)
        $sentencia=$sentencia."TNUMETRAM=".$this->getdecNumeTram().", ";

        $sentencia=substr($sentencia,0,strlen($sentencia)-2);
        $sentencia=$sentencia." where TCODILINE=".$this->getdecCodiLine();

        $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
        $objConexion->desconectar($link);
        return $bolResp;
    }

    // Elimina una fila

```

```

// Devuelve un valor lógico
function eliminar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="delete from linea where TCODILINE='". $this->
>getdecCodiLine()." ";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Método que devuelve un sólo registro. La consulta se hace por
llave primaria
// Los datos se setean en los atributos de la clase
function consultar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TCODILINE, TNOMBLINE, TIDENUSUA, TTIROHORI, TPESOCND, TPESOCADAEISL, TALTUCADE
AISL, TPESOROLD, TTEMPINST, TNUMETRAM from linea where TCODILINE='". $this->
>getdecCodiLine()." ";
    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    //***** setear los atributos con los datos obtenidos *****
    foreach($vecDatos as $vecAux)
    {
        $this->setdecCodiLine($vecAux[0]);
        $this->setstrNombLine($vecAux[1]);
        $this->setstrIdenUsua($vecAux[2]);
        $this->setdecTiroHori($vecAux[3]);
        $this->setdecPesoCond($vecAux[4]);
        $this->setdecPesoCadeAisl($vecAux[5]);
        $this->setdecAltuCadeAisl($vecAux[6]);
        $this->setdecPesoRold($vecAux[7]);
        $this->setdecTempInst($vecAux[8]);
        $this->setdecNumeTram($vecAux[9]);
    }
    //*****
    $objConexion->desconectar($link);
}

```

```

}

//Método que devuelve uno o varios registros según los atributos
seteados.
//Devuelve un vector de objetos.
function buscar()
{
    $objConexion    new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TCODILINE, TNOMBLINE, TIDENUSUA, TTIROHORI, TPESOCOD, TPESOCADEAISL, TALTUCADE
AISL, TPESOROLD, TTEMPINST from linea ";

    /***** validaciones para la opción "where" del query. Sólo
se tomarán los atributos != null *****/
    $numFiltros=0;
    if ($this->getdecCodiLine()!=null)
    {
        # no hay validación por ser el primer filtro
        $sentencia=$sentencia."where TCODILINE=".$this->
getdecCodiLine()." ";
        $numFiltros++;
    }
    if ($this->getstrNombLine()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TNOMBLINE='". $this->
getstrNombLine()." ' ";
        $numFiltros++;
    }
    if ($this->getstrIdenUsua()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TIDENUSUA='". $this->
getstrIdenUsua()." ' ";
        $numFiltros++;
    }
    if ($this->getdecTiroHori()!=null)

```



```

    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TTIROHORI=".$this-
>getdecTiroHori()." ";
        $numFiltros++;
    }
    if ($this->getdecPesoCond()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TPESOCOND=".$this-
>getdecPesoCond()." ";
        $numFiltros++;
    }
    if ($this->getdecPesoCadeAisl()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TPESOCADEAISL=".$this-
>getdecPesoCadeAisl()." ";
        $numFiltros++;
    }
    if ($this->getdecAltuCadeAisl()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TALTUCADEAISL=".$this-
>getdecAltuCadeAisl()." ";
        $numFiltros++;
    }
    if ($this->getdecPesoRold()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TPESOROLD='". $this-
>getdecPesoRold()." ' ";
        $numFiltros++;
    }
    if ($this->getdecTempInst()!=null)

```

```

        {
            if ($numFiltros==0) $sentencia=$sentencia."where ";else
            $sentencia=$sentencia."and ";
            $sentencia=$sentencia."TEMPINST=".$this-
>getdecTempInst()." ";
            $numFiltros++;
        }
        if ($this->getdecNumeTram()!=null)
        {
            if ($numFiltros==0) $sentencia=$sentencia."where ";else
            $sentencia=$sentencia."and ";
            $sentencia=$sentencia."TNUMETRAM=".$this-
>getdecNumeTram()." ";
            $numFiltros++;
        }

//*****
*****

        $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
        //setear los datos obtenidos a los atributos
        $i=0;
        foreach($vecDatos as $vecAux)
        {
            $objLinea = new Linea;
            $objLinea->setdecCodiLine($vecAux[0]);
            $objLinea->setstrNombLine($vecAux[1]);
            $objLinea->setstrIdenUsua($vecAux[2]);
            $objLinea->setdecTiroHori($vecAux[3]);
            $objLinea->setdecPesoCond($vecAux[4]);
            $objLinea->setdecPesoCadeAisl($vecAux[5]);
            $objLinea->setdecAltuCadeAisl($vecAux[6]);
            $objLinea->setdecPesoRold($vecAux[7]);
            $objLinea->setdecTempInst($vecAux[8]);
            $objLinea->setdecNumeTram($vecAux[9]);
            $vecLineas[$i]=$objLinea;
            $i++;
        }
//*****

        $objConexion->desconectar($link);

```

```
        return $vecLineas;
    }

} //fin de la clase
?>
```

B.1.2 Clase Tramo

```
<?php
include_once("../clases/Conexion.cls.php");

class Tramo
{
    var $decCodiLine;
    var $decCodiTram;
    var $strNombTram;
    var $decNumeVano;
    var $decVanoCont;

    function getdecCodiLine()
    {
        return $this->decCodiLine;
    }

    function setdecCodiLine($newdecCodiLine)
    {
        $this->decCodiLine=$newdecCodiLine;
    }

    function getdecCodiTram()
    {
        return $this->decCodiTram;
    }

    function setdecCodiTram($newdecCodiTram)
    {
        $this->decCodiTram=$newdecCodiTram;
    }
}
```

```
function getstrNombTram()
{
    return $this->strNombTram;
}

function setstrNombTram($newstrNombTram)
{
    $this->strNombTram=$newstrNombTram;
}

function getdecNumeVano()
{
    return $this->decNumeVano;
}

function setdecNumeVano($newdecNumeVano)
{
    $this->decNumeVano=$newdecNumeVano;
}

function getdecVanoCont()
{
    return $this->decVanoCont;
}

function setdecVanoCont($newdecVanoCont)
{
    $this->decVanoCont=$newdecVanoCont;
}

// Inserta todos los campos
// Devuelve un valor lógico
function agregar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia= "insert into tramo      (TCODILINE
                                     ,TCODITRAM
                                     ,TNOMBTRAM
```

```

, TNUMEVANO
, TVANOCONT
)
values      (".$this-
>getdecCodiLine()==null?"null":$this->getdecCodiLine())."
, ".$this-
>getdecCodiTram()==null?"null":$this->getdecCodiTram())."
, ".$this->getstrNombTram()."
, ".$this-
>getdecNumeVano()==null?"null":$this->getdecNumeVano())."
, ".$this-
>getdecVanoCont()==null?"null":$this->getdecVanoCont())."
)";

    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Actualiza sólo los campos que han sido seteados
// Devuelve un valor lógico
function actualizar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="update tramo set ";
    if ($this->getstrNombTram()!=null)
    $sentencia=$sentencia."TNOMBTRAM='". $this->getstrNombTram()."', ";
    if ($this->getdecNumeVano()!=null)
    $sentencia=$sentencia."TNUMEVANO=".$this->getdecNumeVano().", ";
    if ($this->getdecVanocont()!=null)
    $sentencia=$sentencia."TVANOCONT=".$this->getdecVanoCont().", ";
    $sentencia=substr($sentencia,0,strlen($sentencia)-2);
    $sentencia=$sentencia." where TCODILINE=".$this-
>getdecCodiLine();
    $sentencia=$sentencia." and TCODITRAM=".$this-
>getdecCodiTram();
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

```

```

}

// Elimina una fila
// Devuelve un valor lógico
function eliminar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="delete from tramo ";
    $sentencia=$sentencia." where TCODILINE='".$this->getdecCodiLine()."'";
    $sentencia=$sentencia." and TCODITRAM='".$this->getdecCodiTram()."'";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Método que devuelve un sólo registro. La consulta se hace por
llave primaria
// Los datos se setean en los atributos de la clase
function consultar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TCODILINE,TCODITRAM,TNOMBTRAM,TNUMEVANO,TVANOCONT from tramo ";
    $sentencia=$sentencia." where TCODILINE='".$this->getdecCodiLine()."'";
    $sentencia=$sentencia." and TCODITRAM='".$this->getdecCodiTram()."'";
    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    /***** setear los atributos con los datos obtenidos *****/
    foreach($vecDatos as $vecAux)
    {
        $this->setdecCodiLine($vecAux[0]);
        $this->setdecCodiTram($vecAux[1]);
        $this->setstrNombTram($vecAux[2]);
        $this->setdecNumeVano($vecAux[3]);
        $this->setdecVanoCont($vecAux[4]);
    }
}

```

```

    }
    //*****
    $objConexion->desconectar($link);
}

//Método que devuelve uno o varios registros según los atributos
seteados.
//Devuelve un vector de objetos.
function buscar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TCODILINE,TCODITRAM,TNOMBTRAM,TNUMEVANO,TVANOCONT from tramo ";

    //***** validaciones para la opcion "where" del query. Sólo
se tomarán los atributos != null *****
    $numFiltros=0;
    if ($this->getdecCodiLine()!=null)
    {
        # no hay validación por ser el primer filtro
        $sentencia=$sentencia."where TCODILINE='". $this-
>getdecCodiLine()."' ";
        $numFiltros++;
    }
    if ($this->getdecCodiTram()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TCODITRAM='". $this-
>getdecCodiTram()."' ";
        $numFiltros++;
    }
    if ($this->getstrNombTram()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TNOMBTRAM='". $this-
>getstrNombTram()."' ";
        $numFiltros++;
    }
}

```

```

    }
    if ($this->getdecNumeVano() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";
        $sentencia = $sentencia."TNUMEVANO='". $this->
>getdecNumeVano()."' ";
        $numFiltros++;
    }
    if ($this->getdecVanoCont() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";
        $sentencia = $sentencia."TVANOCONT='". $this->
>getdecVanoCont()."' ";
        $numFiltros++;
    }

    //*****
    *****

    $vecDatos = $objConexion->ejecutarSelect($sentencia, $link);
    //setear los datos obtenidos a los atributos
    $i = 0;
    foreach($vecDatos as $vecAux)
    {
        $objTramo = new Tramo;
        $objTramo->setdecCodiLine($vecAux[0]);
        $objTramo->setdecCodiTram($vecAux[1]);
        $objTramo->setstrNombTram($vecAux[2]);
        $objTramo->setdecNumeVano($vecAux[3]);
        $objTramo->setdecVanoCont($vecAux[4]);
        $vecTramos[$i] = $objTramo;
        $i++;
    }
    //*****
    $objConexion->desconectar($link);
    return $vecTramos;
}

```



```
}//fin de la clase  
?>
```

B.1.3 Clase Vano

```
<?php  
include_once("../classes/Conexion.cls.php");  
  
class Tramo  
{  
    var $decCodiLine;  
    var $decCodiTram;  
    var $strNombTram;  
    var $decNumeVano;  
    var $decVanoCont;  
  
    function getdecCodiLine()  
    {  
        return $this->decCodiLine;  
    }  
  
    function setdecCodiLine($newdecCodiLine)  
    {  
        $this->decCodiLine=$newdecCodiLine;  
    }  
  
    function getdecCodiTram()  
    {  
        return $this->decCodiTram;  
    }  
  
    function setdecCodiTram($newdecCodiTram)  
    {  
        $this->decCodiTram=$newdecCodiTram;  
    }  
  
    function getstrNombTram()  
    {
```

```
        return $this->strNombTram;
    }

    function setstrNombTram($newstrNombTram)
    {
        $this->strNombTram=$newstrNombTram;
    }

    function getdecNumeVano()
    {
        return $this->decNumeVano;
    }

    function setdecNumeVano($newdecNumeVano)
    {
        $this->decNumeVano=$newdecNumeVano;
    }

    function getdecVanoCont()
    {
        return $this->decVanoCont;
    }

    function setdecVanoCont($newdecVanoCont)
    {
        $this->decVanoCont=$newdecVanoCont;
    }

    // Inserta todos los campos
    // Devuelve un valor lógico
    function agregar()
    {
        $objConexion = new Conexion;
        $link=$objConexion->conectar();
        $sentencia= "insert into tramo      (TCODILINE
                    , TCODITRAM
                    , TNOMBTRAM
                                                , TNUMEVANO
                    , TVANOCONT
```

```

        values      (".( $this-
>getdecCodiLine()==null?"null":$this->getdecCodiLine())."
        ,".( $this-
>getdecCodiTram()==null?"null":$this->getdecCodiTram())."
        ,'".$this->getstrNombTram().'"
        ,".( $this-
>getdecNumeVano()==null?"null":$this->getdecNumeVano())."
        ,".( $this-
>getdecVanoCont()==null?"null":$this->getdecVanoCont())."
        )";

        $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
        $objConexion->desconectar($link);
        return $bolResp;
    }

    // Actualiza sólo los campos que han sido seteados
    // Devuelve un valor lógico
    function actualizar()
    {
        $objConexion = new Conexion;
        $link=$objConexion->conectar();
        $sentencia="update tramo set ";
        if ($this->getstrNombTram() != null)
        $sentencia=$sentencia."TNOMBTRAM='". $this->getstrNombTram()."', ";
        if ($this->getdecNumeVano() != null)
        $sentencia=$sentencia."TNUMEVANO='". $this->getdecNumeVano()."', ";
        if ($this->getdecVanocont() != null)
        $sentencia=$sentencia."TVANOCONT='". $this->getdecVanoCont()."', ";
        $sentencia=substr($sentencia,0,strlen($sentencia)-2);
        $sentencia=$sentencia." where TCODILINE='". $this->
>getdecCodiLine();
        $sentencia=$sentencia." and TCODITRAM='". $this->
>getdecCodiTram();
        $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
        $objConexion->desconectar($link);
        return $bolResp;
    }

    // Elimina una fila

```

```

// Devuelve un valor lógico
function eliminar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="delete from tramo ";
    $sentencia=$sentencia." where TCODILINE='".$this->getdecCodiLine()."'";
    $sentencia=$sentencia." and TCODITRAM='".$this->getdecCodiTram()."'";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Método que devuelve un sólo registro. La consulta se hace por
llave primaria
// Los datos se setean en los atributos de la clase
function consultar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TCODILINE,TCODITRAM,TNOMBTRAM,TNUMEVANO,TVANOCONT from tramo ";
    $sentencia=$sentencia." where TCODILINE='".$this->getdecCodiLine()."'";
    $sentencia=$sentencia." and TCODITRAM='".$this->getdecCodiTram()."'";
    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    //***** setear los atributos con los datos obtenidos *****
    foreach($vecDatos as $vecAux)
    {
        $this->setdecCodiLine($vecAux[0]);
        $this->setdecCodiTram($vecAux[1]);
        $this->setstrNombTram($vecAux[2]);
        $this->setdecNumeVano($vecAux[3]);
        $this->setdecVanoCont($vecAux[4]);
    }
    //*****
    $objConexion->desconectar($link);
}

```

```

}

//Método que devuelve uno o varios registros según los atributos
seteados.
//Devuelve un vector de objetos.
function buscar()
{
    $objConexion    new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TCODILINE,TCODITRAM,TNOMBTRAM,TNUMEVANO,TVANOCONT from tramo ";

    //***** validaciones para la opcion "where" del query. Sólo
se tomarán los atributos != null *****
    $numFiltros=0;
    if ($this->getdecCodiLine()!=null)
    {
        # no hay validación por ser el primer filtro
        $sentencia=$sentencia."where TCODILINE='". $this-
>getdecCodiLine()."' ";
        $numFiltros++;
    }
    if ($this->getdecCodiTram()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TCODITRAM='". $this-
>getdecCodiTram()."' ";
        $numFiltros++;
    }
    if ($this->getstrNombTram()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TNOMBTRAM='". $this-
>getstrNombTram()."' ";
        $numFiltros++;
    }
    if ($this->getdecNùmeVano()!=null)
    {

```

```

        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TNUMEVANO='". $this-
>getdecNumeVano()." ";
        $numFiltros++;
    }
    if ($this->getdecVanoCont()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TVANOCONT='". $this-
>getdecVanoCont()." ";
        $numFiltros++;
    }

    //*****
    *****

    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    //setear los datos obtenidos a los atributos
    $i=0;
    foreach($vecDatos as $vecAux)
    {
        $objTramo = new Tramo;
        $objTramo->setdecCodiLine($vecAux[0]);
        $objTramo->setdecCodiTram($vecAux[1]);
        $objTramo->setstrNombTram($vecAux[2]);
        $objTramo->setdecNumeVano($vecAux[3]);
        $objTramo->setdecVanoCont($vecAux[4]);
        $vecTramos[$i]=$objTramo;
        $i++;
    }
    //*****
    $objConexion->desconectar($link);
    return $vecTramos;
}

} //fin de la clase
?>

```

B.1.4 Clase GrupoUsuario

```
<?php
include_once("../clases/Conexion.cls.php");

class GrupoUsuario
{
    var $strCodiGrupUsua;
    var $strDescGrupUsua;
    var $strEstaGrupUsua;

    function getstrCodiGrupUsua()
    {
        return $this->strCodiGrupUsua;
    }

    function setstrCodiGrupUsua($newstrCodiGrupUsua)
    {
        $this->strCodiGrupUsua=$newstrCodiGrupUsua;
    }

    function getstrDescGrupUsua()
    {
        return $this->strDescGrupUsua;
    }

    function setstrDescGrupUsua($newstrDescGrupUsua)
    {
        $this->strDescGrupUsua=$newstrDescGrupUsua;
    }

    function getstrEstaGrupUsua()
    {
    }

    .
    {
    }
}
```

```

// Inserta todos los campos
// Devuelve un valor lógico
function agregar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia= "insert into grupo_usuario (TCODIGRUPUSUA
                ,TDESCGRUPUSUA
                ,TESTAGRUPUSUA
                )
                values      ('".$this->
>strCodiGrupUsua()."
                , '".$this->strDescGrupUsua()."
                , '".$this->strEstaGrupUsua()."
                )";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Actualiza sólo los campos que han sido seteados
// Devuelve un valor lógico
function actualizar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="update grupo_usuario set ";
    if ($this->getstrDescGrupUsua() != null)
    $sentencia=$sentencia."TDESCGRUPUSUA='".$this->getstrDescGrupUsua()."
";
    if ($this->getstrEstaGrupUsua() != null)
    $sentencia=$sentencia."TESTAGRUPUSUA='".$this->getstrEstaGrupUsua()."
";
    $sentencia=substr($sentencia,0,strlen($sentencia)-2);
    $sentencia=$sentencia." where TCODIGRUPUSUA='".$this->
>getstrCodiGrupUsua()."
";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

```



```

// Elimina una fila
// Devuelve un valor lógico
function eliminar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="delete from grupo_usuario ";
    $sentencia=$sentencia." where TCODIGRUPUSUA='". $this->getstrCodiGrupUsua()." "'";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Método que devuelve un sólo registro. La consulta se hace por llave primaria
// Los datos se setean en los atributos de la clase
function consultar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select TCODIGRUPUSUA, TDESCGRUPUSUA, TESTAGRUPUSUA from grupo_usuario ";
    $sentencia=$sentencia." where TCODIGRUPUSUA='". $this->getstrCodiGrupUsua()." "'";
    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    //***** setear los atributos con los datos obtenidos *****
    foreach($vecDatos as $vecAux)
    {
        $this->setstrCodiGrupUsua($vecAux[0]);
        $this->setstrDescGrupUsua($vecAux[1]);
        $this->setstrEstaGrupUsua($vecAux[2]);
    }
    //*****
    $objConexion->desconectar($link);
}

//Método que devuelve uno o varios registros según los atributos seteados.

```

```

//Devuelve un vector de objetos.
function buscar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select TCODIGRUPUSUA,TDESCGRUPUSUA,TESTAGRUPUSUA
from grupo_usuario ";

    /***** validaciones para la opcion "where" del query. Sólo
se tomarán los atributos != null *****/
    $numFiltros=0;
    if ($this->getstrCodiGrupUsua()!=null)
    {
        # no hay validación por ser el primer filtro
        $sentencia=$sentencia."where TCODIGRUPUSUA='". $this-
>getstrCodiGrupUsua()."' ";
        $numFiltros++;
    }
    if ($this->getstrDescGrupUsua()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TDESCGRUPUSUA='". $this-
>getstrDescGrupUsua()."' ";
        $numFiltros++;
    }
    if ($this->getstrEstaGrupUsua()!=null)
    {
        if ($numFiltros==0) $sentencia=$sentencia."where ";else
        $sentencia=$sentencia."and ";
        $sentencia=$sentencia."TESTAGRUPUSUA='". $this-
>getstrEstaGrupUsua()."' ";
        $numFiltros++;
    }

    /*******
*****

    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    //setear los datos obtenidos a los atributos

```

```

        $i=0;
        foreach($vecDatos as $vecAux)
        {
            $objLinea = new Linea;
            $objLinea->setstrCodiGrupUsua($vecAux[0]);
            $objLinea->setstrDescGrupUsua($vecAux[1]);
            $objLinea->setstrEstaGrupUsua($vecAux[2]);
            $vecLinea[$i]=$objLinea;
            $i++;
        }
        //*****
        $objConexion->desconectar($link);
        return $vecLinea;
    }

} //fin de la clase
?>

```

B.1.5 Clase Usuario

```

<?php
include_once("../clases/Conexion.cls.php");

class Usuario
{
    var $strIdenUsua;
    var $strPassUsua;
    var $strNombUsua;
    var $strApelPate;
    var $strApelMate;
    var $strEstaUsua;
    var $strCodiGrup;
    var $strUsuaCrea;
    var $datFechCrea;
    var $strUsuaModi;
    var $datFechModi;
}

```

```
function getstrIdenUsua()
{
    return $this->strIdenUsua;
}

function setstrIdenUsua($newstrIdenUsua)
{
    $this->strIdenUsua=$newstrIdenUsua;
}

function getstrPassUsua()
{
    return $this->strPassUsua;
}

function setstrPassUsua($newstrPassUsua)
{
    $this->strPassUsua=$newstrPassUsua;
}

function getstrNombUsua()
{
    return $this->strNombUsua;
}

function setstrNombUsua($newstrNombUsua)
{
    $this->strNombUsua=$newstrNombUsua;
}

function getstrApelPate()
{
    return $this->strApelPate;
}

function setstrApelPate($newstrApelPate)
{
    $this->strApelPate=$newstrApelPate;
}
```

```
function getstrApelMate()
{
    return $this->strApelMate;
}

function setstrApelMate($newstrApelMate)
{
    $this->strApelMate=$newstrApelMate;
}

function getstrEstaUsua()
{
    return $this->strEstaUsua;
}

function setstrEstaUsua($newstrEstaUsua)
{
    $this->strEstaUsua=$newstrEstaUsua;
}

function getstrCodiGrup()
{
    return $this->strCodiGrup;
}

function setstrCodiGrup($newstrCodiGrup)
{
    $this->strCodiGrup=$newstrCodiGrup;
}

function getstrUsuaCrea()
{
    return $this->strUsuaCrea;
}

function setstrUsuaCrea($newstrUsuaCrea)
{
    $this->strUsuaCrea=$newstrUsuaCrea;
}
```

```
function getdatFechCrea()
{
    return $this->datFechCrea;
}

function setdatFechCrea($newdatFechCrea)
{
    $this->datFechCrea=$newdatFechCrea;
}

function getstrUsuaModi()
{
    return $this->strUsuaModi;
}

function setstrUsuaModi($newstrUsuaModi)
{
    $this->strUsuaModi=$newstrUsuaModi;
}

function getdatFechModi()
{
    return $this->datFechModi;
}

function setdatFechModi($newdatFechModi)
{
    $this->datFechModi=$newdatFechModii;
}

// Inserta todos los campos
// Devuelve un valor lógico
function agregar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia= "insert into USUARIO (TIDENUSUA
                ,TPASSUSUA
                ,TNOMBUSUA
```

```

        ,TAPELPATE
        ,TAPELMATE
        ,TESTAUSUA
        ,TCODIGRUP
        ,TUSUACREA
        ,TFECHCREA
        ,TUSUAMODI
        ,TFECHMODI
    )
    values ('".$this->getstrIdenUsua()."
        ,".$this->getstrPassUsua()."
        ,".$this->getstrNombUsua()."
        ,".$this->getstrApelPate()."
        ,".$this->getstrApelMate()."
        ,".$this->getstrEstaUsua()."
        ,".$this->getstrCodiGrup()."
        ,".$this->getstrUsuaCrea()."
        ,".$this->getdatFechCrea()."
        ,".$this->getstrUsuaModi()."
        ,".$this->getdatFechModi()."
    )";

    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Actualiza sólo los campos que han sido seteados
// Devuelve un valor lógico
function actualizar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="update USUARIO set ";
    if ($this->getstrPassUsua() != null)
$sentencia=$sentencia."TPASSUSUA=".$this->getstrPassUsua().", ";
    if ($this->getstrNombUsua() != null)
$sentencia=$sentencia."TNOMBUSUA=".$this->getstrNombUsua().", ";
    if ($this->getstrApelPate() != null)
$sentencia=$sentencia."TAPELPATE=".$this->getstrApelPate().", ";

```

```

        if ($this->getstrApelMate() != null)
$sentencia=$sentencia."TAPELMATE=".$this->getstrApelMate().", ";
        if ($this->getstrEstaUsua() != null)
$sentencia=$sentencia."TESTAUSUA=".$this->getstrEstaUsua().", ";
        if ($this->getstrCodiGrup() != null)
$sentencia=$sentencia."TCODIGRUP=".$this->getstrCodiGrup().", ";
        if ($this->getstrUsuaCrea() != null)
$sentencia=$sentencia."TUSUACREA=".$this->getstrUsuaCrea().", ";
        if ($this->getdatFechCrea() != null)
$sentencia=$sentencia."TFECHCREA=".$this->getdatFechCrea().", ";
        if ($this->getstrUsuaModi() != null)
$sentencia=$sentencia."TUSUAMODI=".$this->getstrUsuaModi().", ";
        if ($this->getdatFechModi() != null)
$sentencia=$sentencia."TFECHMODI=".$this->getdatFechModi().", ";
        $sentencia=substr($sentencia,0,strlen($sentencia)-2);
        $sentencia=$sentencia." where TIDENUSUA='".$this->getstrIdenUsua()."'";
        $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
        $objConexion->desconectar($link);
        return $bolResp;
    }

// Elimina una fila
// Devuelve un valor lógico
function eliminar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="delete from USUARIO ";
    $sentencia=$sentencia." where TIDENUSUA='".$this->getstrIdenUsua()."'";
    $bolResp=$objConexion->ejecutarSentencia($sentencia,$link);
    $objConexion->desconectar($link);
    return $bolResp;
}

// Método que devuelve un sólo registro. La consulta se hace por
llave primaria
// Los datos se setean en los atributos de la clase
function consultar()

```



```

{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TIDENUSUA,TPASSUSUA,TNOMBUSUA,TAPELPATE,TAPELMATE,TESTAUSUA,TCODIGRUP,TUS
UACREA,TFECHCREA,TUSUAMODI,TFECHMODI from USUARIO ";
    $sentencia=$sentencia." where TIDENUSUA='". $this-
>getstrIdenUsua()." ";
    $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
    //***** setear los atributos con los datos obtenidos *****
    foreach($vecDatos as $vecAux)
    {
        $this->setstrIdenUsua($vecAux[0]);
        $this->setstrPassUsua($vecAux[1]);
        $this->setstrNombUsua($vecAux[2]);
        $this->setstrApelPate($vecAux[3]);
        $this->setstrApelMate($vecAux[4]);
        $this->setstrEstaUsua($vecAux[5]);
        $this->setstrCodiGrup($vecAux[6]);
        $this->setstrUsuaCrea($vecAux[7]);
        $this->setdatFechCrea($vecAux[8]);
        $this->setstrUsuaModi($vecAux[9]);
        $this->setdatFechModi($vecAux[10]);
    }
    //*****
    $objConexion->desconectar($link);
}

//Método que devuelve uno o varios registros según los atributos
seteados.
//Devuelve un vector de objetos.
function buscar()
{
    $objConexion = new Conexion;
    $link=$objConexion->conectar();
    $sentencia="select
TIDENUSUA,TPASSUSUA,TNOMBUSUA,TAPELPATE,TAPELMATE,TESTAUSUA,TCODIGRUP,TUS
UACREA,TFECHCREA,TUSUAMODI,TFECHMODI from USUARIO ";

```

```

        //***** validaciones para la opcion "where" del query. Sólo
se tomarán los atributos != null *****
        $numFiltros=0;
        if ($this->getstrIdenUsua() != null)

            # no hay validación por ser el primer filtro
                $sentencia=$sentencia."where TIDENUSUA='". $this-
>getstrIdenUsua()."' ";
                $numFiltros++;
            }
        if ($this->getstrPassUsua() != null)
        {
            if ($numFiltros==0) $sentencia=$sentencia."where ";else
            $sentencia=$sentencia."and ";
            $sentencia=$sentencia."TPASSUSUA='". $this-
>getstrPassUsua()."' ";
            $numFiltros++;
        }
        if ($this->getstrNombUsua() != null)
        {
            if ($numFiltros==0) $sentencia=$sentencia."where ";else
            $sentencia=$sentencia."and ";
            $sentencia=$sentencia."TNOMBUSUA='". $this-
>getstrNombUsua()."' ";
            $numFiltros++;
        }
        if ($this->getstrApelPate() != null)
        {
            if ($numFiltros==0) $sentencia=$sentencia."where ";else
            $sentencia=$sentencia."and ";
            $sentencia=$sentencia."TAPELPATE='". $this-
>getstrApelPate()."' ";
            $numFiltros++;
        }
        if ($this->getstrApelMate() != null)
        {
            if ($numFiltros==0) $sentencia=$sentencia."where ";else
            $sentencia=$sentencia."and ";
            $sentencia=$sentencia."TAPELMATE='". $this-
>getstrApelMate()."' ";

```

```

        $numFiltros++;
    }
    if ($this->getstrEstaUsua() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";
        $sentencia = $sentencia."TESTAUSUA=".$this-
>getstrEstaUsua()." ";
        $numFiltros++;
    }
    if ($this->getstrCodiGrup() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";
        $sentencia = $sentencia."TCODIGRUP=".$this-
>getstrCodiGrup()." ";
        $numFiltros++;
    }
    if ($this->getstrUsuaCrea() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";
        $sentencia = $sentencia."TUSUACREA='". $this-
>getstrUsuaCrea()."' ";
        $numFiltros++;
    }
    if ($this->getdatFechCrea() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";
        $sentencia = $sentencia."TFECHCREA='". $this-
>getdatFechCrea()."' ";
        $numFiltros++;
    }

    if ($this->getstrUsuaModi() != null)
    {
        if ($numFiltros == 0) $sentencia = $sentencia."where "; else
        $sentencia = $sentencia."and ";

```

```

                $sentencia=$sentencia."TUSUAMODI='". $this-
>getstrUsuaModi()." ";
                $numFiltros++;
            }

            if ($this->getdatFechModi() != null)
            {
                if ($numFiltros == 0) $sentencia=$sentencia."where "; else
                $sentencia=$sentencia."and ";
                $sentencia=$sentencia."TFECHMODI='". $this-
>getdatFechModi()." ";
                $numFiltros++;
            }

        }

        //*****
        *****

        $vecDatos=$objConexion->ejecutarSelect($sentencia,$link);
        //setear los datos obtenidos a los atributos
        $i=0;
        foreach($vecDatos as $vecAux)
        {
            $objUsuario = new Usuario;
            $objUsuario->setstrIdenUsua($vecAux[0]);
            $objUsuario->setstrPassUsua($vecAux[1]);
            $objUsuario->setstrNombUsua($vecAux[2]);
            $objUsuario->setstrApelPate($vecAux[3]);
            $objUsuario->setstrApelMate($vecAux[4]);
            $objUsuario->setstrEstaUsua($vecAux[5]);
            $objUsuario->setstrCodiGrup($vecAux[6]);
            $objUsuario->setstrUsuaCrea($vecAux[7]);
            $objUsuario->setdatFechCrea($vecAux[8]);
            $objUsuario->setstrUsuaModi($vecAux[9]);
            $objUsuario->setdatFechModi($vecAux[10]);
            $vecUsuario[$i]=$objUsuario;

            $i++;
        }
    }
}

```

```

//*****

$objConexion->desconectar($link);
return $vecUsuario;
}

function validarUsuario($login,$password)
{
    $this->setstrIdenUsua($login);
    $this->setstrPassUsua($password);
    $bolRpta=false;
    if ($this->getstrIdenUsua() != null && $this-
>getstrPassUsua() != null)
    {
        $vecUsuario=$this->buscar();
        foreach($vecUsuario as $objDatosUsuario)
        {
            if ($objDatosUsuario->getstrIdenUsua() != null)
            {
                $bolRpta=true;
            }
        }
    }
    return $bolRpta;
}

} //fin de la clase
?>

```

B.1.6 Clase Conexión

```

<?php

require ("../include/parametrosBD.inc");

class Conexion
{
    var $strServidorBD=SERVIDOR_BD;
    var $strUsuarioBD=USUARIO_BD;

```

```

var $strPasswordBD=PASSWORD_BD;
var $strNombreBD=NOMBRE_BD;

function conectar()
{
    if (!(($link=mysql connect($this->strServidorBD,$this->strUsuarioBD,$this->strPasswordBD)))
    {
        echo "Error conectando a la base de datos.";
        exit();
    }
    if (!mysql_select_db($this->strNombreBD,$link))
    {
        echo "Error seleccionando la base de datos.";
        exit();
    }
    return $link;
}

function desconectar($link)
{
    mysql_close($link);
}

function ejecutarSelect($query,$link)
{
    $result=mysql_query($query,$link);
    $numColumnas=mysql_num_fields($result);

    if (mysql_num_rows($result)==0)//cuando el query devuelve 0
    filas, se devuelve un vector con valores nulos
    {
        $j=0;
        for ($j=0;$j<$numColumnas;$j++)
        {
            $vecAux[$j]=null;
        }
        $vecDatos[0]=$vecAux;
    }
    else

```

```

        {
            $i=0;
            while($row = mysql_fetch_array($result))
            {
                $j=0;
                for ($j=0;$j<$numColumnas;$j++)
                {
                    $vecAux[$j]=$row[$j];
                }
                $vecDatos[$i]=$vecAux;
                $i++;
            }
        }

        mysql_free_result($result);
        return $vecDatos;
    }

function ejecutarSentencia($query,$link)
{
    $bolResp=mysql_query($query,$link);
    return $bolResp;
}
}
?>

```

B.1.7 Clase Util

```

<?php
include_once("../clases/Conexion.cls.php");

class Util
{
    //Método que devuelve un vector con las opciones asignadas a
    determinado tipo de usuario
    function obtenerOpciones($tipo)
    {
        $cod_opc1="1000";
        $nom_opc1="Ingreso de Datos";
    }
}

```

```

$url_opc1="../paginas/s_ingreso.php?tipo=1";
$des_opc1="Pantalla para buscara la persona que solicita la
atención";

```

```

$cod_opc2="2000";
$nom_opc2="Actualización de <br> Datos";
$url_opc2="../paginas/s_actualizar.php?tipo=1";
$des_opc2="Pantalla de Bandeja de entrada del Supervisor";

```

```

$cod_opc3="3000";
$nom_opc3="Consulta";
$url_opc3="../paginas/s_bandejaTecnico.php";
$des_opc3="Pantalla de Bandeja de entrada del Técnico";

```

```

$cod_opc4="4000";
$nom_opc4="Seguimiento";
$url_opc4="../paginas/s_seguimientoSupervisor.php";
$des_opc4="Pantalla del Seguimiento del Requerimiento";

```

```

$cod_opc5="5000";
$nom_opc5="Reportes";
$url_opc5="../paginas/reporteSupervisor.php";
$des_opc5="Pantalla del Reporte del Supervisor";

```

```

$cod_opc6="6000";
$nom_opc6="Reportes";
$url_opc6="../paginas/reporteTécnico.php";
$des_opc6="Pantalla del Reporte del Técnico";

```

```

switch ($tipo)

```

```

{

```

```

case "01":

```

```

    $vecAux["codigo"]=$cod_opc1;

```

```

    $vecAux["url"]=$url_opc1;

```

```

    $vecAux["nombre"]=$nom_opc1;

```

```

    $vecDatos[0]=$vecAux;

```

```

    $vecAux["codigo"]=$cod_opc2;

```

```

    $vecAux["url"]=$url_opc2;

```

```

    $vecAux["nombre"]=$nom_opc2;

```

```

    $vecDatos[1]=$vecAux;

```



```

    $vecAux["codigo"]=$cod_opc3;
    $vecAux["url"]=$url_opc3;
    $vecAux["nombre"]=$nom_opc3;
    $vecDatos[2]=$vecAux;

    break;
case "02":
    $vecAux["codigo"]=$cod_opc2;
    $vecAux["url"]=$url_opc2;
    $vecAux["nombre"]=$nom_opc2;
    $vecDatos[0]=$vecAux;
    $vecAux["codigo"]=$cod_opc4;
    $vecAux["url"]=$url_opc4;
    $vecAux["nombre"]=$nom_opc4;
    $vecDatos[1]=$vecAux;
    $vecAux["codigo"]=$cod_opc5;
    $vecAux["url"]=$url_opc5;
    $vecAux["nombre"]=$nom_opc5;
    $vecDatos[2]=$vecAux;
    break;
case "03":
    $vecAux["codigo"]=$cod_opc3;
    $vecAux["url"]=$url_opc3;
    $vecAux["nombre"]=$nom_opc3;
    $vecDatos[0]=$vecAux;
    $vecAux["codigo"]=$cod_opc4;
    $vecAux["url"]=$url_opc4;
    $vecAux["nombre"]=$nom_opc4;
    $vecDatos[1]=$vecAux;
    $vecAux["codigo"]=$cod_opc6;
    $vecAux["url"]=$url_opc6;
    $vecAux["nombre"]=$nom_opc6;
    $vecDatos[2]=$vecAux;
    break;
default :
    $vecAux["codigo"]=null;
    $vecAux["url"]=null;
    $vecAux["nombre"]=null;
    $vecDatos[0]=$vecAux;
}

```

```

        return $vecDatos;
    }
    //Método que devuelve la secuencia de un determinado atributo
    function obtenerSecuencia($campo)
    {
        $objConexion = new Conexion;
        $link=$objConexion->conectar();
        $select="select TDESCSECU,TVALOSECU from secuencia where
trim(TDESCSECU)='".$trim($campo)."'";
        $vecDatos=$objConexion->ejecutarSelect($select,$link);
        $valor=$vecDatos[0][1];
        $nuevoValor=$valor+1;
        $sentencia="update secuencia set TVALOSECU='".$nuevoValor.'"
where trim(TDESCSECU)='".$trim($campo)."'";
        $objConexion->ejecutarSentencia($sentencia,$link);
        $objConexion->desconectar($link);
        return $valor;
    }
    //Método que devuelve un vector con los tipos de notificación
    function obtenerTipoNotificacion()
    {
        $vecAux["valor"]="E";
        $vecAux["texto"]="Email";
        $vecTipoNotificacion[0]=$vecAux;
        $vecAux["valor"]="C";
        $vecAux["texto"]="Mansaje Celular";
        $vecTipoNotificacion[1]=$vecAux;
        $vecAux["valor"]="B";
        $vecAux["texto"]="Mensaje Beeper";
        $vecTipoNotificacion[2]=$vecAux;
        return $vecTipoNotificacion;
    }
    //Método que devuelve un vector con los números de los dias
    function obtenerDia()
    {
        for($i=1;$i<=31;$i++)
        {
            $vecAux["valor"]=str_pad($i,2,"0",STR_PAD_LEFT);
            $vecAux["texto"]=str_pad($i,2,"0",STR_PAD_LEFT);
            $vecDia[$i-1]=$vecAux;
        }
    }

```

```
    }  
    return $vecDia;  
}  
  
//Método que devuelve un vector con los nombres de los meses  
function obtenerMes()  
{  
    $vecAux["valor"]="01";  
    $vecAux["texto"]="Enero";  
    $vecMes[0]=$vecAux;  
    $vecAux["valor"]="02";  
    $vecAux["texto"]="Febrero";  
    $vecMes[1]=$vecAux;  
    $vecAux["valor"]="03";  
    $vecAux["texto"]="Marzo";  
    $vecMes[2]=$vecAux;  
    $vecAux["valor"]="04";  
    $vecAux["texto"]="Abril";  
    $vecMes[3]=$vecAux;  
    $vecAux["valor"]="05";  
    $vecAux["texto"]="Mayo";  
    $vecMes[4]=$vecAux;  
    $vecAux["valor"]="06";  
    $vecAux["texto"]="Junio";  
    $vecMes[5]=$vecAux;  
    $vecAux["valor"]="07";  
    $vecAux["texto"]="Julio";  
    $vecMes[6]=$vecAux;  
    $vecAux["valor"]="08";  
    $vecAux["texto"]="Agosto";  
    $vecMes[7]=$vecAux;  
    $vecAux["valor"]="09";  
    $vecAux["texto"]="Septiembre";  
    $vecMes[8]=$vecAux;  
    $vecAux["valor"]="10";  
    $vecAux["texto"]="Octubre";  
    $vecMes[9]=$vecAux;  
    $vecAux["valor"]="11";  
    $vecAux["texto"]="Noviembre";  
    $vecMes[10]=$vecAux;  
    $vecAux["valor"]="12";
```

```

    $vecAux["texto"]="Diciembre";
    $vecMes[11]=$vecAux;

    return $vecMes;
}
//Método que devuelve un vector con los nombres de los meses
function obtenerAnio()
{
    $vecAux["valor"]="2004";
    $vecAux["texto"]="2004";
    $vecAnio[0]=$vecAux;
    $vecAux["valor"]="2005";
    $vecAux["texto"]="2005";
    $vecAnio[1]=$vecAux;

    return $vecAnio;
}
function obtenerHora()
{
    for($i=1;$i<=12;$i++)
    {
        $vecAux["valor"]=str_pad($i,2,"0",STR_PAD_LEFT);
        $vecAux["texto"]=str_pad($i,2,"0",STR_PAD_LEFT);
        $vecHora[$i-1]=$vecAux;
    }
    return $vecHora;
}
function obtenerMinuto()
{
    for($i=1;$i<=59;$i++)
    {
        $vecAux["valor"]=str_pad($i,2,"0",STR_PAD_LEFT);
        $vecAux["texto"]=str_pad($i,2,"0",STR_PAD_LEFT);
        $vecMinuto[$i-1]=$vecAux;
    }
    return $vecMinuto;
}
function obtenerMeridiano()
{
    $vecAux["valor"]="AM";

```

```

        $vecAux["texto"]="am";
        $vecMeridiano[0]=$vecAux;
        $vecAux["valor"]="PM";
        $vecAux["texto"]="pm";
        $vecMeridiano[1]=$vecAux;
        return $vecMeridiano;
    }
function obtenerPrimUltDiaSemana($anio,$mes,$dia)
{
    $diaSemana=date("w",mktime(0,0,0,$mes,$dia,$anio));
    $fechaIni=date("Y-m-d H:i:s",mktime(0,0,0,$mes,$dia-$
$diaSemana+1,2004));
    $fechaFin=date("Y-m-d H:i:s",mktime(23,59,59,$mes,$dia-$
$diaSemana+1+5,2004));
    $vecPrimUlt["fechaIni"]=$fechaIni;
    $vecPrimUlt["fechaFin"]=$fechaFin;
    $vecPrimUlt["timestampIni"]=mktime(0,0,0,$mes,$dia-$
$diaSemana+1,2004);
    $vecPrimUlt["timestampFin"]=mktime(23,59,59,$mes,$dia-$
$diaSemana+1+5,2004);
    return $vecPrimUlt;
}
}
?>

```

B.1.8 Clase Calculo

```

<?php
include_once("../clases/Linea.cls.php");
include_once("../clases/Tramo.cls.php");
include_once("../clases/Vano.cls.php");

class Calculo
{
    // Método que realiza el cálculo del clipping offset
    // @codiLine Código de la línea.
    // @codiTram Código del tramo.

```

```

function clippingOffset($objLinea,$objTramo)
{

    $objLinea->consultar();
    $objTramo->consultar();

    $objVano = new Vano;
    $objVano->setdecCodiLine($objLinea->getdecCodiLine());
    $objVano->setdecCodiTram($objTramo->getdecCodiTram());
    $vecVanos = $objVano->buscar();

    $ tiroHori = $objLinea->getdecTiroHori();
    $pesoCond = $objLinea->getdecPesoCond();
    $pesoCadeAisl = $objLinea->getdecPesoCadeAisl();
    $altuCadeAisl = $objLinea->getdecAltuCadeAisl();
    $pesoRold = $objLinea->getdecPesoRold();
    $numeVano = $objTramo->getdecNumeVano();
    $vanoCont = $objTramo->getdecVanoCont();
    $paraCate = $ tiroHori / $pesoCond;

    //PRIMERA PARTE:
    //Se recorren los vanos para obtener la longitud del vano ($a)
y la diferencia de altura($h)
    for ($i = 1;$i <= $numeVano;$i++)
    {
        $objVanoAux = $vecVanos[$i-1];
        $a[$i] = $objVanoAux->getdecDistHori();
        $h[$i] = $objVanoAux->getdecDifeAltu();
    }

    //VANOS A LA IZQUIERDA DEL VANO DE CONTROL
    for ($i = $vanoCont;$i >= 1;$i--)
    {
        //Se calculan los tiros en el vértice y los parámetros
de la catenaria en cada vano
        if ($i != $vanoCont)
        {
            $C[$i] = $this-
>parametroCatenariaIzq($a[$i],$h[$i],$T[$i],$pesoCond);

```

```

        $To[$i] = $pesoCond * $C[$i];
    }
    else
    {
        $To[$i] = $tiroHori;
        $C[$i] = $To[$i] / $pesoCond;
    }

    //Se calcula la longitud del cable como si estuvieran al
mismo nivel
    $L[$i] = 2 * $C[$i] * sinh($a[$i] / (2 * $C[$i]));

    //Se calcula el valor de x en el punto medio del vano
    $Xm[$i] = $C[$i] * $this->asinh($h[$i] / $L[$i]);

    //se calcula el valor de x en la torre anterior
    $Xi[$i] = $Xm[$i] - ($a[$i] / 2);

    //Se calcula el tiro en la torre anterior
    $T[$i-1] = $To[$i] * cosh($Xi[$i] / $C[$i]);
}

//VANOS A LA DERECHA DEL VANO DE CONTROL
for ($i = $vanoCont;$i <= $numeVano;$i++)
{
    //Se calculan los tiros en el vértice y los parámetros
de la catenaria en cada vano
    if ($i != $vanoCont)
    {
        $C[$i] = $this-
>parametroCatenariaDer($a[$i], $h[$i], $T[$i], $pesoCond);
        $To[$i] = $pesoCond * $C[$i];
    }

    //Se calcula la longitud del cable como si estuvieran al
mismo nivel
    $L[$i] = 2 * $C[$i] * sinh($a[$i] / (2 * $C[$i]));

    //Se calcula el valor de x en el punto medio del vano
    $Xm[$i] = $C[$i] * $this->asinh($h[$i] / $L[$i]);

```

```

//Se calcula el valor de x en la torre anterior
$Xf[$i] = $Xm[$i] + ($a[$i] / 2);

//Se calcula el tiro en la torre anterior
$T[$i+1] = $To[$i] * cosh($Xf[$i] / $C[$i]);
}

//SEGUNDA PARTE
for ($i = 1;$i <= $numeVano;$i++)
{
    $Lant = $C[$i] * sinh(($Xm[$i] + ($a[$i] / 2)) /
$C[$i]);

    if ($i != $numeVano)
    {
        $Lpos = -$C[$i+1] * sinh(($Xm[$i+1] - ($a[$i+1] /
2))/ $C[$i+1]);

        $Lw[$i] = $Lant + $Lpos;
        $d[$i] = atan(($To[$i+1] - $To[$i]) /
(($pesoCadeAisl / 2) + ($pesoCond * $Lw[$i]) + $pesoRold));
    }
    else
    {
        $Lw[$i] = $Lant;
        $d[$i] = "0";
    }
}

$Dl[$numeVano] = "0";
for ($i = 1;$i <= $numeVano-1;$i++)
{
    $De[$i] = $altuCadeAisl * (sin($d[$i+1]) - sin($d[$i]));
    $Dh[$i] = $altuCadeAisl * (cos($d[$i]) - cos($d[$i+1]));

    $Lpol = sqrt(pow(2 * $C[$i] * sinh(($a[$i] + $De[$i]) /
(2 * $C[$i])),2) + pow($h[$i] + $Dh[$i],2));
    $Lgra = sqrt(pow(2 * $paraCate * sinh($a[$i] / (2 *
$paraCate)),2) + pow($h[$i],2));
}

```



```

        $Dl[$i] = ($Lpol - $Lgra);
        //$Dl[$i] = -$Dl[$i]; // para que la cadena de
aisladores quede horizontal entonces hay
                                // que corregir el valor negativo de la
diferencia de longitud
                                // en poleas - longitud en grapas

    }

//TERCERA PARTE
//Se recorren los vanos para asignarle el cálculo del clipping
offset
for ($i = 1;$i <= $numeVano;$i++)
{
    $objVanoAux = $vecVanos[$i-1];
    $objVanoAux->setdecDeltLong($Dl[$i]);
    $objVanoAux->actualizar();
}
}

// Método que realiza el cálculo del parametro de la catenaria de
los vanos a la izquierda del vano de control.
// @a : Longitud del vano.
// @h : Diferencia de alturas.
// @T : Tiro.
// @w : Peso del conductor
function parametroCatenariaIzq($a,$h,$T,$w)
{
    $k = 1 + 0.5 * pow($this->asinh($h/$a),2);
    $aux = (1/$k) * (($T/$w) - ($h/2));
    $parCat = 0.5 * ($aux + sqrt(pow($aux,2) - (($a *
sqrt(pow($a,2) + pow($h,2)))/(2 * $k))));
    return $parCat;
}

// Método que realiza el cálculo del parametro de la catenaria de
los vanos a la derecha del vano de control.

```

```

// @a : Longitud del vano.
// @h : Diferencia de alturas.
// @T : Tiro.
function parametroCatenariaDer($a,$h,$T,$w)
{
    $k = 1 + 0.5 * pow($this->asinh($h/$a),2);
    $aux  (1/$k) * (($T/$w) + ($h/2));
    $parCat = 0.5 * ($aux + sqrt(pow($aux,2) - (($a *
sqrt(pow($a,2) + pow($h,2)))/(2 * $k))));
    return $parCat;
}

// Método que realiza el cálculo del arcoseno hyperbólico ya que el
PHP no implemente esta función bajo Windows.
// @x : Valor para aplicar a la función
function asinh($x)
{
    $asinh = log($x + sqrt($x * $x +1));
    return $asinh;
}

} //fin de la clase
?>

```

BIBLIOGRAFÍA

1. IEEE, “ TRANSMISION & DISTRIBUTION”,
IEEE, Inc - EEUU, Octubre 1995.
2. IEEE, “Norma IEEE Std. 524-1980”,
IEEE, Inc - EEUU, 1980.
3. Juan Bautista Rios, “Líneas de Transmisión de Potencia Volumen I: Aspectos
Mecánicos y Conductores (Preedición 2004)”,
IT Solutions Digitalización Empresarial S.A.C- Perú, 2004.
4. Stig Sæther Bakken, “PHP Manual”,
PHP Documentation Group-EEUU, 2003.
5. MySQL AB, “MySQL 3.23, 4.0, 4.1 Reference Manual”,
MySQL Documentation Group-EEUU, 2005.