Universidad Politécnica
de Madrid

ÁRBOL MÍNIMO DE STEINER
SMT

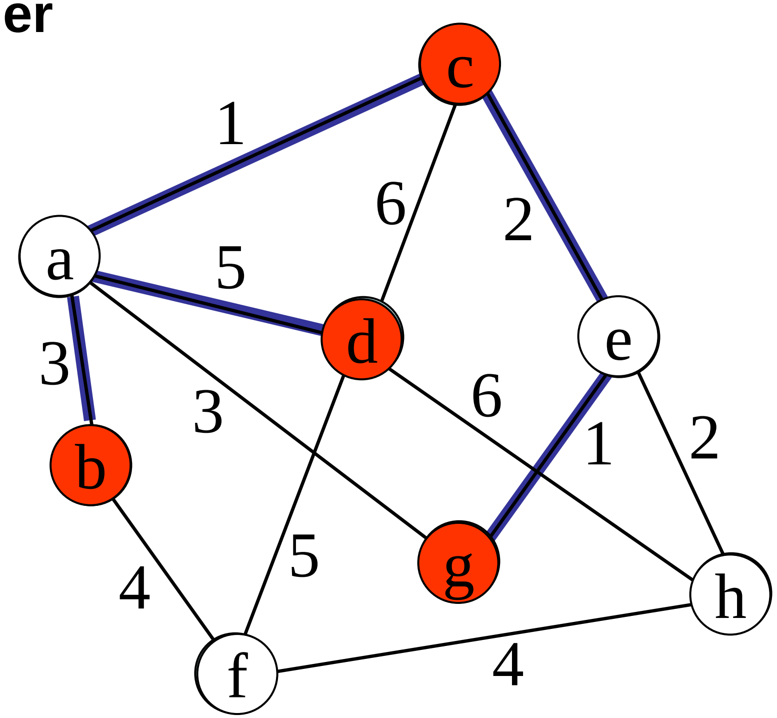
Gregorio Hernández
UPM

Optimización Combinatoria, 2018

STEINER MINIMAL TREES

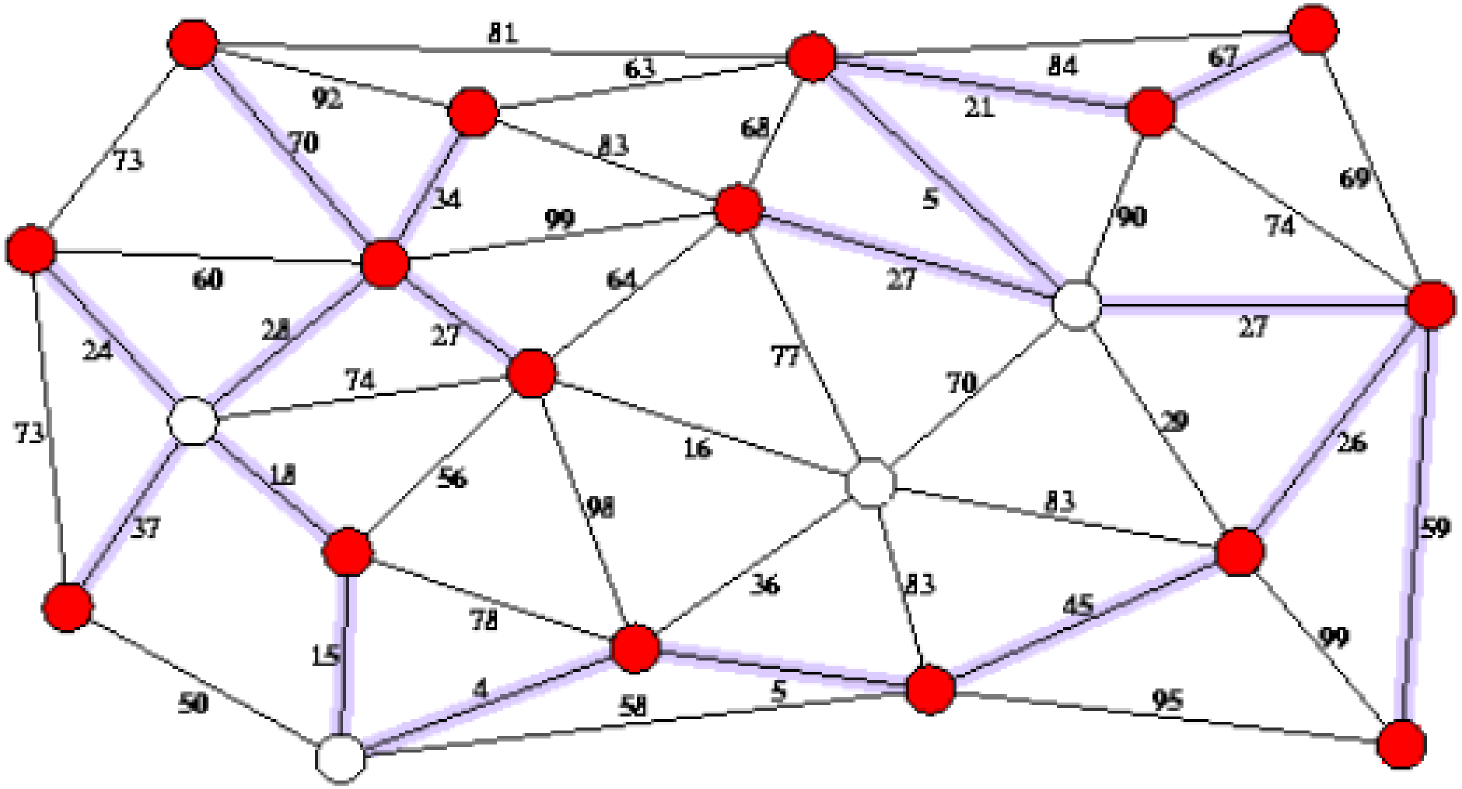
Los puntos adicionales que se añaden a una configuración geométrica para mejorar sus propiedades reciben el sobrenombre de puntos de Steiner

Un árbol generador de un grafo conecta **TODOS** sus vértices.
Un **árbol de Steiner** sólo conecta un subconjunto de vértices (**terminales**) pudiendo utilizar otros (**vértices de Steiner** o no terminales, blancos) para minimizar el peso del árbol así construido.



STEINER MINIMAL TREES

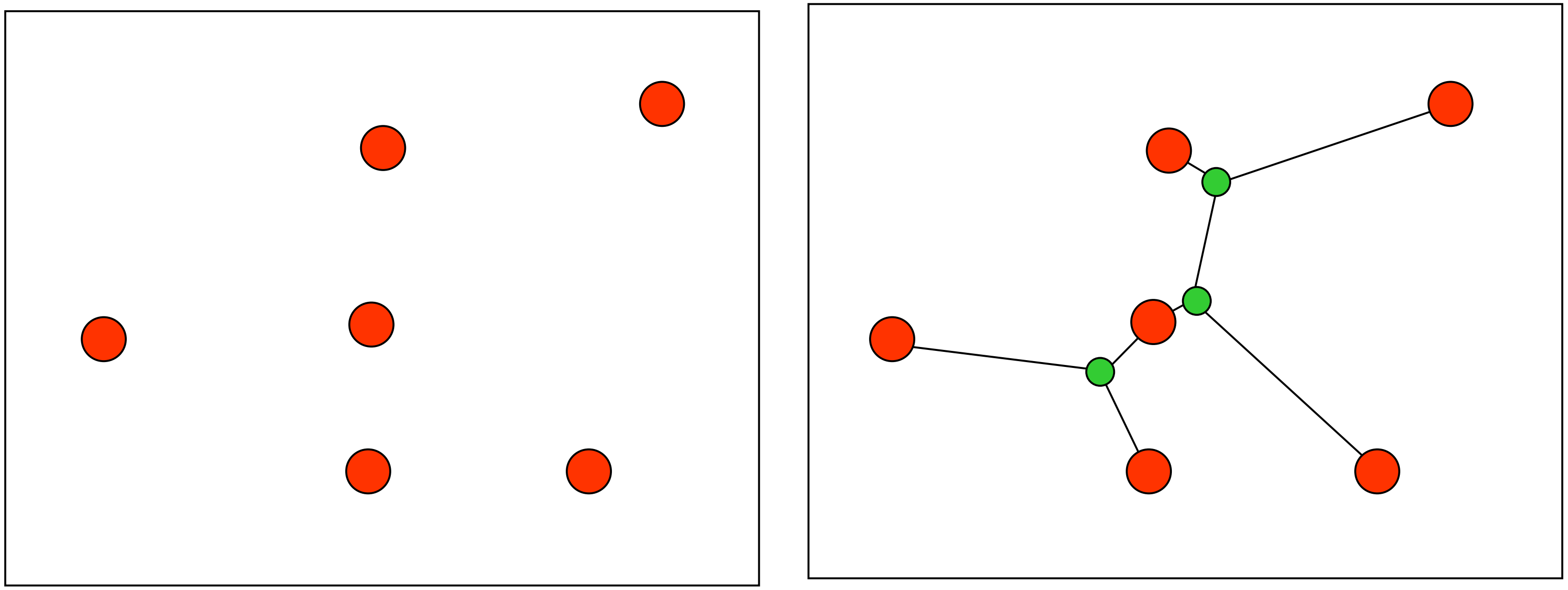
Los puntos adicionales que se añaden a una configuración geométrica para mejorar sus propiedades reciben el sobrenombre de puntos de Steiner



EUCLIDEAN STEINER MINIMAL TREE

Versión euclídea, **"Problema del conector mínimo"**

Dado un conjunto de puntos P del plano, encontrar un conjunto S de puntos (puntos de Steiner) tales que el árbol generador mínimo euclídeo de P + S tenga mínima longitud

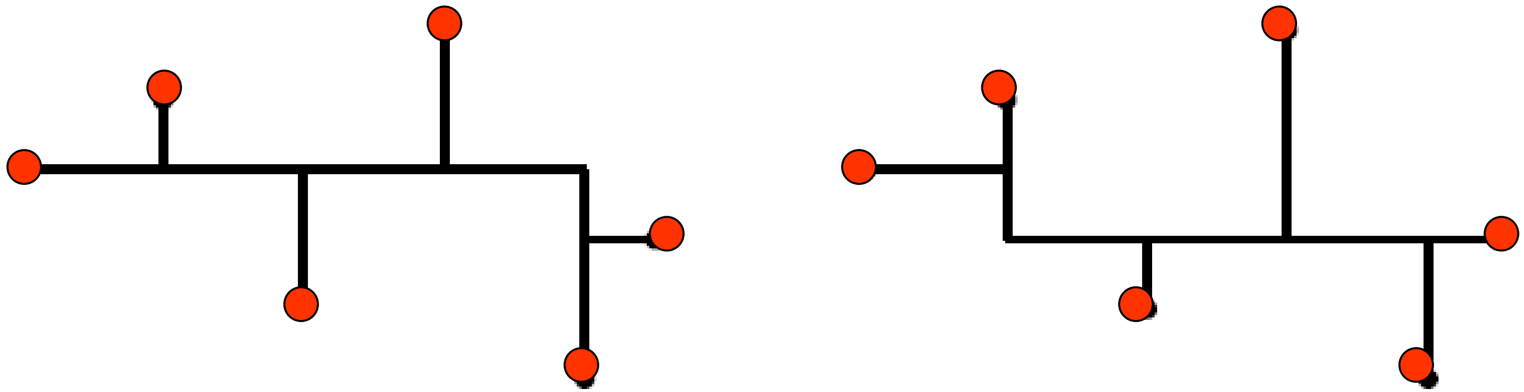


RECTILINEAR EUCLIDEAN STEINER MINIMAL TREE

"Problema del conector mínimo"

con métrica L_1 (o Manhattan)

Dado un conjunto de puntos P del plano, encontrar un conjunto S de puntos (puntos de Steiner) tales que el árbol generador mínimo de P + S tenga mínima longitud con la métrica L_1



(Graph) STEINER MINIMAL TREES

$G = (V,A)$ $L \subset V$, conjunto de terminales

Se busca un árbol T de peso mínimo que contenga TODOS los vértices de L

StMT(G) Problema NP-completo

Algoritmo 2-aproximado

(1) Cierre métrico G_L
(2) Árbol generator de G_L $MST(G_L) = T_L$
(3) Reemplazar cada arista de T_L por el camino mínimo en G

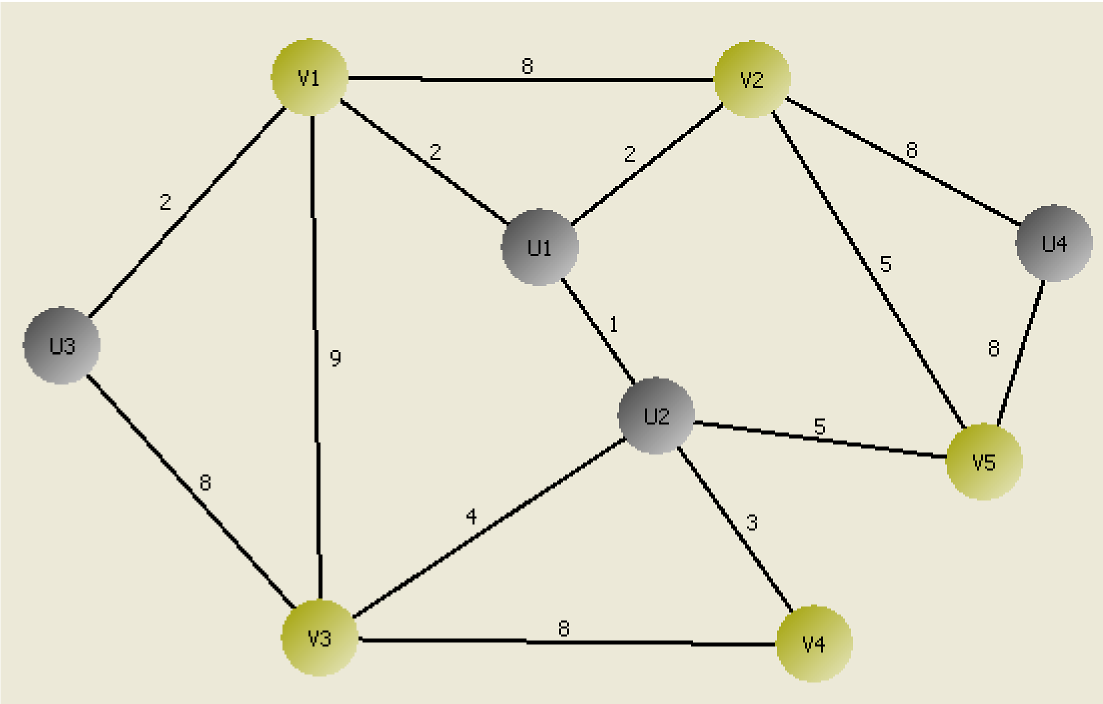
http://www.dma.fi.upm.es/personal/gregorio/grafos/web/arboles_optimos/Principal.html

StMT(G) Algoritmo 2-aprox

Algoritmo 2-aproximado

(1) Cierre métrico G_L

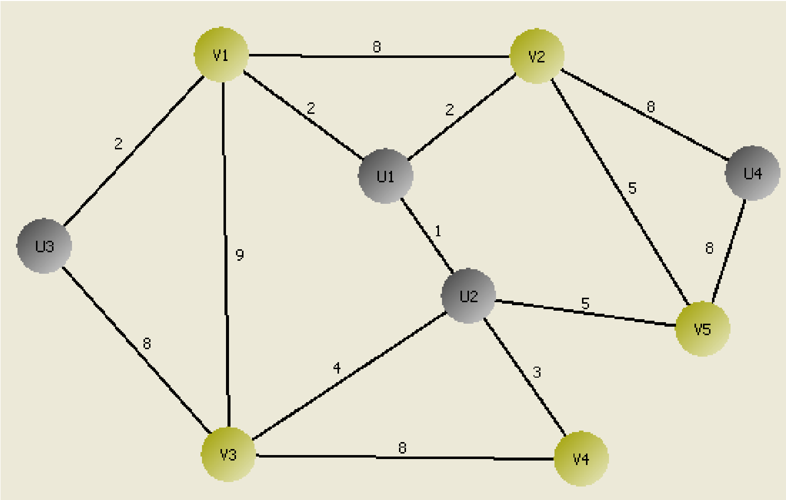
Se construye el grafo completo sobre L, K_L, considerando que el peso de cada arista es el peso del camino mínimo en G

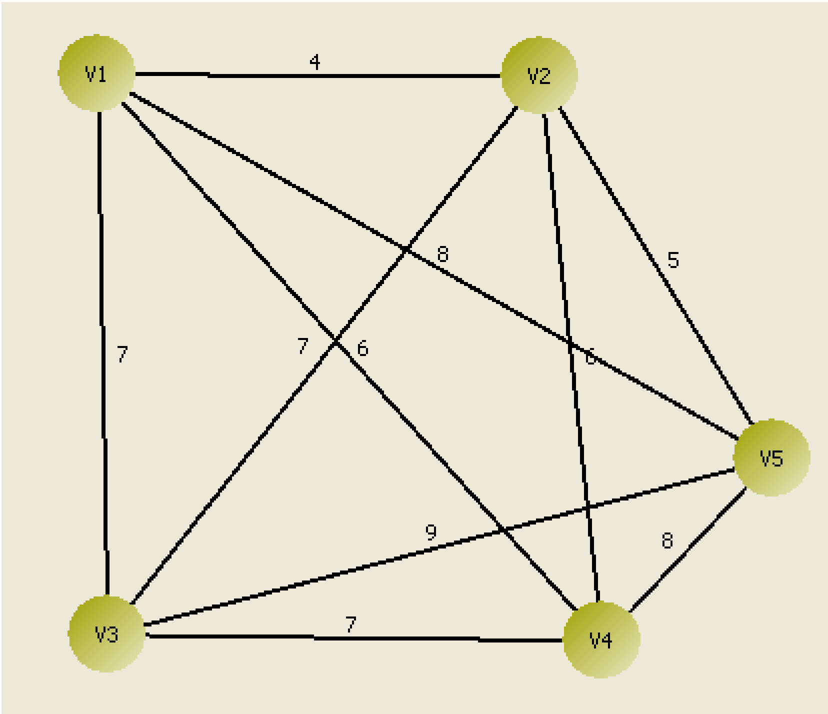


Algoritmo 2-aproximado

(1) Cierre métrico G_L

Se construye el grafo completo sobre L, K_L, considerando que el peso de cada arista es el peso del camino mínimo en G

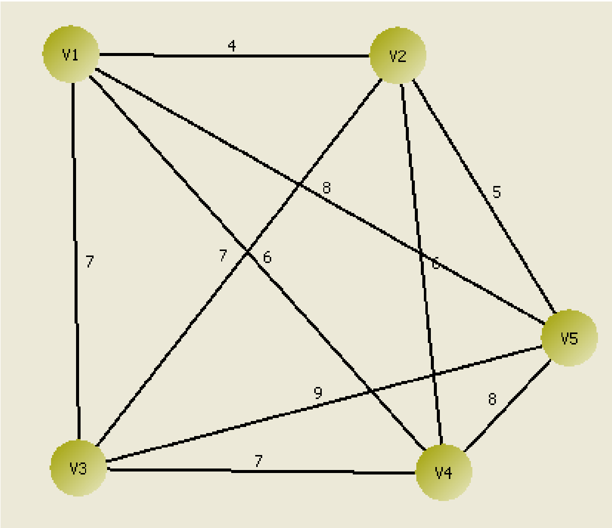


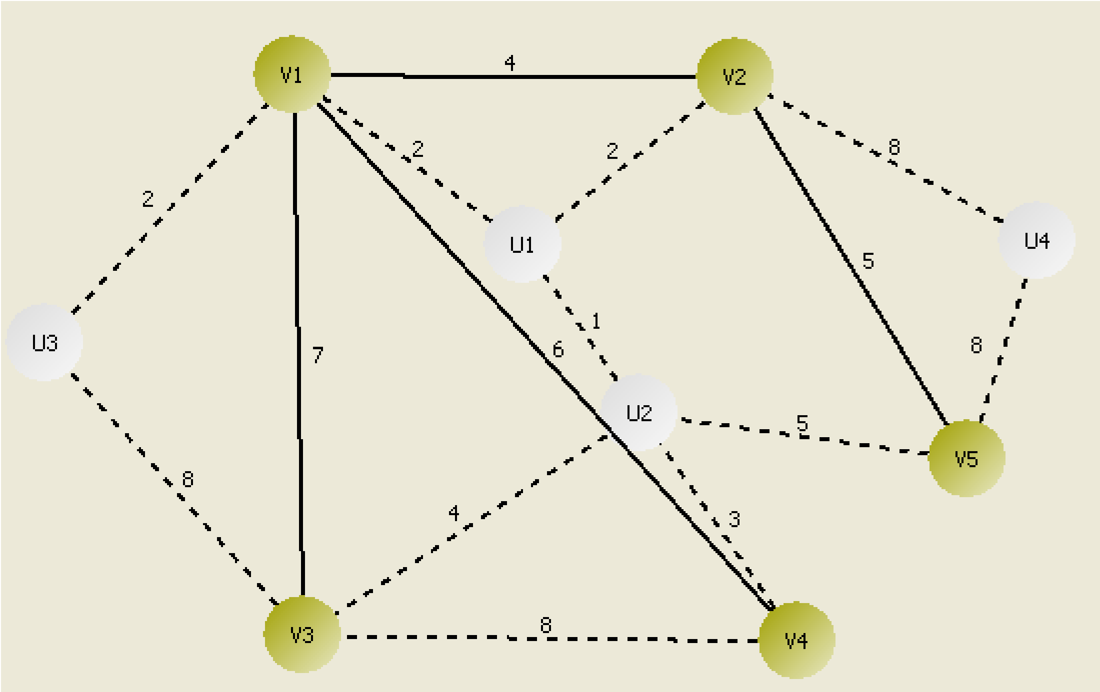


StMT(G) Algoritmo 2-aprox

Algoritmo 2-aproximado

(2) Construir T_L el árbol generador mínimo de G_L

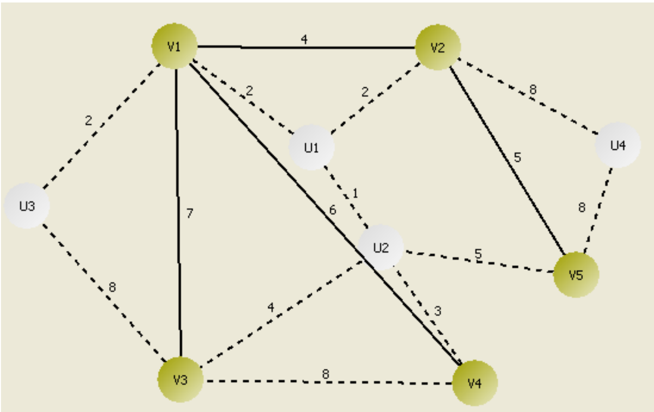


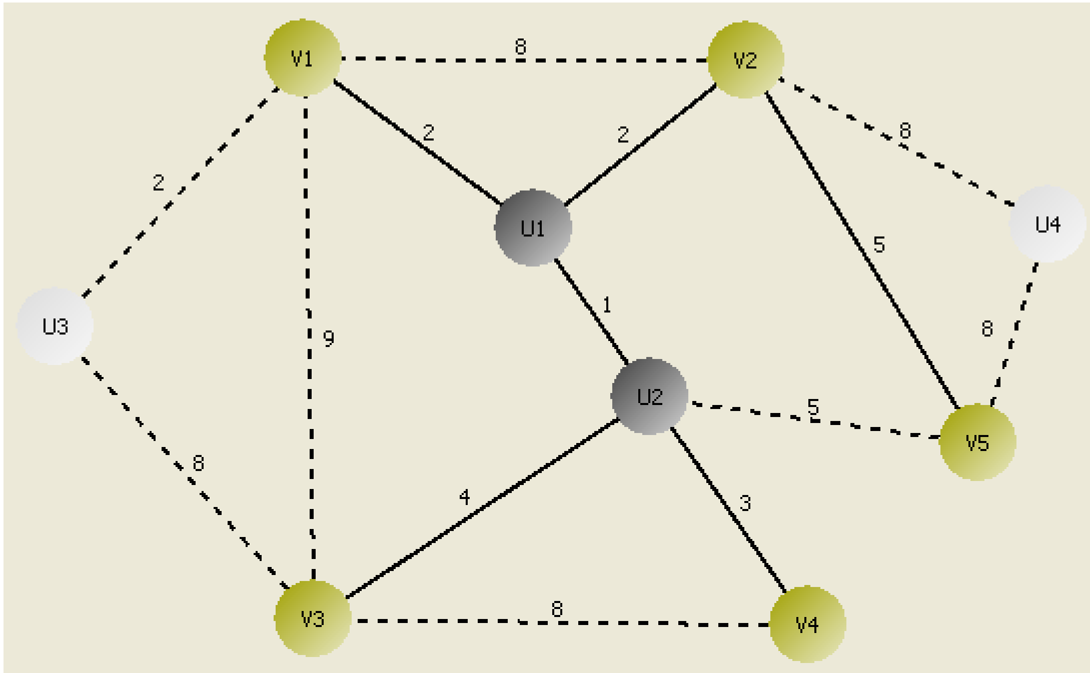


StMT(G) Algoritmo 2-aprox

Algoritmo 2-aproximado

(3) Reemplazar cada arista de T_L por el camino mínimo en G





StMT(G) Algoritmo 2-aprox

Factor de aproximación

Sea T el árbol obtenido en el algoritmo, $w(T) \leq w(T_L)$
y $T^* = SMT(G,L)$

Consideramos el grafo X obtenido al duplicar las aristas de T*
$w(X) = 2w(T^*)$

Como X visita todos los terminales $w(X) \geq w(TSP(G_L)) \geq w(T_L)$
donde $TSP(G_L)$ es el recorrido óptimo del viajante para G_L

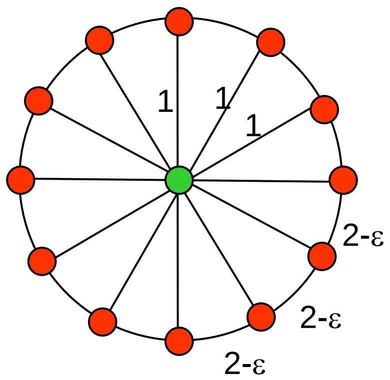
Por tanto, $w(T) \leq w(T_L) \leq 2w(T^*)$

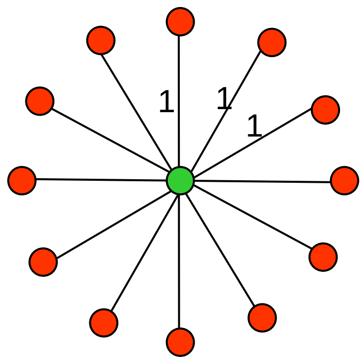
StMT(G)   Algoritmo 2-aprox

Factor de aproximación

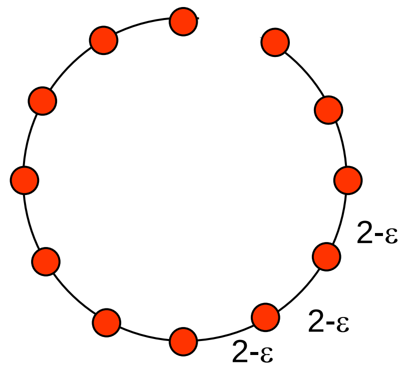
El factor de aproximación es ajustado

Consideramos la rueda W_k con pesos 1 en los radios y 2 - ε en el ci clo.
L = {vértices del ciclo}





StMT(G,L)



T_L

w(StMT) = k $w(T_L) = 2(k – 1)$

$\boxed{\text{Factor } 2 – 2/k}$

StMT(G) Algoritmo 2-aprox

Complejidad del algoritmo

(1)	Cierre métrico G_L	$O(n\|L\|^2)$
(2)	$MST(G_L)$	$O(q + n\log n)$
(3)	Sustituir cada arista de T_L por camino mínimo	$O(q)$

StMT(G) Otros algoritmos

Zelikovsky, 1993, razón $11/6 \approx 1.83$

Robins, Zelikovsky, 2000, razón 1.55

Idea: Utlizar los vértices terminales

Steiner Ratio

Razón entre el peso del MST(G_L) y del StMT(G,L)

$$\boxed{\rho = \frac{w(SMT(G,L))}{w(MST(G_L))}}$$

En grafos $\rho = 1/2$

En el caso euclídeo $\rho = \frac{\sqrt{3}}{2}$

Conjetura Gilbert, Pollack, '68
demostrada Du, Hwang, '92

En métrica L_1 2/3 , Hwang '76