



From spanners to multipath spanners

Quentin Godfroy

► To cite this version:

Quentin Godfroy. From spanners to multipath spanners. Data Structures and Algorithms [cs.DS]. Université Sciences et Technologies - Bordeaux I, 2012. English. NNT : . tel-00984938

HAL Id: tel-00984938

<https://theses.hal.science/tel-00984938>

Submitted on 28 Apr 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

PRÉSENTÉE À

L'UNIVERSITÉ BORDEAUX-I

ÉCOLE DOCTORALE MATHÉMATIQUE ET INFORMATIQUE

Par Quentin GODFROY

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ : Informatique

Des spanneurs aux spanneurs multichemins
From spanners to multipath spanners

Sous la direction de : Cyril Gavoille

Thèse soutenue le: 29 novembre 2012

Devant la commission d'examen formée de :

M. Prosenjit BOSE, Professeur à l'Université de Carleton (Canada);

M. Victor CHEPOI, Professeur à l'Université d'Aix-Marseille-II;

M. Cyril GAVOILLE, Professeur à l'Université Bordeaux-I;

M. Nicolas HANUSSE, Directeur de Recherche au CNRS;

M. Laurent VIENNOT, Directeur de Recherche à l'INRIA.

Résumé Cette thèse traite de l'étude des spanneurs multichemins, comme extension des spanneurs de graphes classiques.

Un spanneur H d'un graphe G est un sous-graphe couvrant tel que pour toute paire de sommets du graphe $a, b \in V(G)$ la distance dans le spanneur $d_H(a, b)$ n'est pas trop étirée par rapport à la distance dans le graphe d'origine $d_G(a, b)$. Ainsi il existe un facteur d'étirement (α, β) tel que pour tout $a, b \in V(G)$, $d_H(a, b) \leq \alpha d_G(a, b) + \beta$.

Motivés par des considérations de routage à plusieurs chemins et après la remarque que le concept de spanneur peut être étendu à toute métrique « non décroissante », nous introduisons la notion de spanneur multichemins.

Après une introduction au domaine, nous parlerons des résultats obtenus concernant d'une part les spanneurs multichemins arêtes disjoints et d'autre part les spanneurs multichemins sommets disjoints.

Abstract This thesis deals with multipath spanners, as an extension of classical graph spanners.

A spanner H of a graph G is a spanning subgraph such that for any pair of vertices $a, b \in V(G)$ the distance measured in the spanner $d_H(a, b)$ isn't too much stretched compared to the distance measured in the original graph $d_G(a, b)$. As such there exists a stretch factor (α, β) such that for all $a, b \in V(G)$, $d_H(a, b) \leq \alpha d_G(a, b) + \beta$.

Motivated by multipath routing and after noting that the concept of spanner can be extended to any “non decreasing” metric, we introduce the notion of multipath spanner.

After an introduction to the topic, we will show the results obtained. The first part is devoted to edge-disjoint multipath spanners. The second part is devoted to vertex-disjoint spanners.

Remerciements

Une thèse ne se fait pas sans directeur de thèse. Je remercie donc Cyrille Gavaille d’avoir accepté de diriger la mienne. Je le remercie aussi pour sa patience, sa gentillesse et sa disponibilité lors de nos rencontres hebdomadaires.

De même qu’une thèse ne se fait pas sans directeur de thèse, elle ne se fait pas sans jury de thèse ni relecteurs. Je remercie Porsenjit Bose et Victor Chepoi d’avoir accepté de relire ma thèse et d’avoir fait le déplacement de loin pour la soutenance. Je remercie aussi Laurent Viennot et Nicolas Hanusse d’avoir accepté d’être membres du jury.

Il n’est pas concevable de faire une thèse dans de bonnes conditions sans laboratoire d’accueil. Je remercie donc le LaBRI de m’avoir hébergé pendant ces trois ans dans d’excellentes conditions matérielles et scientifiques.

Je remercie aussi l’Agence Nationale de la Recherche, sans laquelle je n’aurais pas eu mon financement

La recherche ne se fait pas dans son coin. Je remercie donc tous les gens qui ont participé à mes recherches. Je pense notamment à Laurent Viennot pour les nombreuses discussions tout au long de ma thèse. Je pense aussi à David Peleg, pour m’avoir accueilli au sein de laboratoire de mathématiques de l’Institut Weizmann. Enfin, je remercie tous les participants des groupes de travail “Graphes et Applications” et “Algorithmique Distribuée”.

Je remercie aussi les secrétaires du LaBRI qui m’ont aidé lors de mes nombreux déplacements. Je pense notamment à Elia Meyre grâce à laquelle j’ai pu organiser plusieurs missions compliquées.

Je remercie David Madore pour son soutien constant et pour avoir sû réussi à maintenir une vie de couple avec un turbo-thésard longue distance.

Je remercie aussi mes parents pour leur soutien durant ces trois ans.

Enfin, je remercie la SNCF pour m’avoir transporté sur plus de 430 000 km pendant ces trois ans de thèse.

Contents

1	Présentation	8
1	Presentation	11
2	Introduction	14
2.1	Topic introduction	14
2.1.1	Graphs and distance	14
2.1.2	Spanners	15
2.1.3	Spanner extensions	17
2.2	Multipath spanners	19
2.2.1	Motivation	19
2.2.2	Multipath distance	20
2.2.3	Multipath spanners	23
2.3	Our results	24
2.3.1	Edge-disjoint multipath spanners	24
2.3.2	Vertex-disjoint multipath spanners	24
3	Edge-disjoint multipath spanners	26
3.1	Preliminaries	26
3.1.1	Multipath distance	26
3.1.2	Multipath edge-disjoint spanners	28
3.2	Results	28
3.2.1	Edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanner	28
3.2.2	Lower bounds for edge-disjoint multipath spanners	30
3.2.3	An unweighted edge-disjoint 2-multipath 3-spanner	31
3.2.4	A 2-multipath $(2, 8W)$ -spanner	38
3.2.5	From edge-fault tolerant spanners to multipath spanners	43
4	Vertex-disjoint multipath spanners	50
4.1	Preliminaries	50
4.1.1	Multipath graph metric	50

4.1.2	Vertex-disjoint multipath spanners	51
4.2	Results	52
4.2.1	A first remark concerning fault-tolerant spanners	52
4.2.2	From fault-tolerant spanners to multipath spanners	53
4.2.3	More faults lead to better stretch	60
4.2.4	Towards an additive bi-path spanner	67
5	Conclusion & perspectives	78
5.1	Review of the results	78
5.1.1	Edge-disjoint multipath spanners	78
5.1.2	Vertex-disjoint multipath spanners	79
5.2	Perspectives	80
5.2.1	Improvements	80
5.2.2	Non-decreasing graph metrics	81

List of Figures

2.1	Edge e removal increases the distance $d_G(u, v)$ from 1 to $2k + 1$. . .	17
2.2	This subgraph can be obtain as a combination of four paths, and hence is a 4-path.	21
2.3	The 4-path shown in Figure 2.2 contains only three edge-disjoint paths, but is not in itself a 3-path. Is shown here one of the 3-path contained in the original 4-path.	21
2.4	The process into which an edge $e = x - y$ is replaced by a directed path, ensuring it is crossed only once. For the flow procedure only the edge $x' \rightarrow y'$ has capacity 1 and weight $\omega(e)$. The other are set to 0.	22
2.5	Process by which an undirected graph is transformed into a suitable directed graph to compute the vertex-disjoint p -multipath distance ${}^v d^p$	23
3.1	Any cut separating u from w has capacity at least 3	27
3.2	Proof of Lemma 1.	32
3.3	The different cases for the rules are shown here for an edge from P_1^i . In blue are shown the edges and paths belonging to H_1 , and in red those from H_2	34
3.4	Two possible cases.	36
3.5	A weighted graph G with $d_G^2(x, y) = d$ showing that the stretch analysis in the proof of Theorem 3 is tight. The 2-shortest-path tree rooted at u spans all the edges but (x, y) . We have $d_T^2(u, x) = d_T^2(u, y) = d + 4$, and $d_T^2(x, y) = 2d + 8 - 2\varepsilon$	43
3.6	Definition of M_j , e_h^j and e_l^j . M is the reunion of all M_j	45
4.1	${}^v d^2(u, v)$ and ${}^v d^2(v, w)$ are finite whereas ${}^v d^p(u, w) = \infty$	51
4.2	H is a 1-fault-tolerant $2s$ -spanner, but of stretch at least n when considered as a 2-multipath spanner for the ${}^v d^2$ graph metric. . . .	53

4.3	A subgraph P_H constructed by adding paths between u and v with at most $s = 5$ edges and with $p = 2$. Removing any vertex leaves a path of at most 5 edges, so $\kappa_5(u, v) > 1$. However, there aren't two vertex-disjoint paths from u to v of at most 5 edges, so $\kappa_5(u, v) > \mu_5(u, v)$. Observe that $\mu_6(u, v) = \kappa_5(u, v) = 2$	58
4.4	Illustration of the sets $\mathcal{B}_h, \mathcal{B}_l$ ($p = 2$) and the path M	63
4.5	Process by which X is transformed into X'	69
4.6	Proof of Lemma 19: a cycle of 6 hops exists between w and uv in G_i	73
4.7	Proof of Lemma 21: the two cases for the simple paths.	74
4.8	The two cases for Q^1 and Q^2 . The big dots represent the paths' unused portions.	75

Chapter 1

Présentation

Ce mémoire traite de l'étude des spanneurs multichemins, comme extension des spanneurs de graphes classiques.

Un spanneur (« spanner » en anglais) H d'un graphe G est un sous-graphe couvrant (ie., $V(H) = V(G)$) tel que pour toute paire de sommets du graphe $a, b \in V(G)$, la distance (mesurée comme la longueur d'un plus court chemin) dans le spanneur $d_H(a, b)$ n'est pas trop étirée par rapport à la distance dans le graphe d'origine $d_G(a, b)$. Ainsi, il existe un facteur d'étirement (« stretch » en anglais) (α, β) tel que pour tout $a, b \in V(G)$, $d_H(a, b) \leq \alpha \cdot d_G(a, b) + \beta$. Un spanneur tel que $\beta = 0$ est dit multiplicatif.

Les spanneurs ont été introduits initialement par Peleg et Schäffer [PS89] pour les graphes non valués, puis étendus aux graphes valués [ADD⁺93]. Il existe une abondante littérature sur les spanneurs dont on peut lire un survol par Pettie [Pet07].

Motivés par des considérations de routage à plusieurs chemins et après la remarque que le concept de spanneur peut être étendu à toute métrique « non décroissante » (c'est à dire que toute suppression d'arêtes ne peut faire qu'augmenter la distance), nous introduisons la notion de spanneur multichemin, défini à l'aide de la métrique multichemin. Après une introduction au domaine, nous parlons des résultats obtenus sur les spanneurs multichemins arêtes disjoints dans le Chapitre 3 et les résultats sur les spanneurs multichemins sommets disjoints Chapitre 4.

Ce mémoire est organisé comme suit, chaque chapitre étant prévu pour être largement indépendant des chapitres qui le précèdent :

Chapitre 2 : Introduction. Dans cette partie, après une brève introduction aux concepts nécessaires de la théorie des graphes, nous rappelons le contexte de la théorie des spanneurs, ainsi que quelques résultats.

Le routage multichemins étant intrinsèquement lié à la tolérance aux fautes, nous présentons les spanneurs tolérants aux fautes.

Enfin, nous présentons l'extension proposée dans ce mémoire : les spanneurs multichemins. Nous commençons par définir la métrique multichemins à l'aide du concept des p -chemins qui existe sous deux formes, chemins arêtes disjoints et chemins, puis nous définissons les spanneurs multichemins s'appuyant sur cette métrique.

Chapitre 3 : Edge-disjoint multipath spanners. Dans ce chapitre, après un rappel des définitions de la métrique multichemins arêtes disjoints s'appuyant sur les p -chemins, nous définissons formellement les spanneurs p -multichemins arêtes disjoints (Définition 3).

Sont ensuite exposés les résultats obtenus pendant la thèse : d'abord nous constatons (Théorème 1) qu'un algorithme simple et connu utilisant les spanneurs classiques permet d'obtenir des spanneurs p -multichemins arêtes disjoints avec un étirement multiplié par p par rapport à l'étirement du spanneur d'origine. Ensuite, il est montré (Proposition 2) que les bornes inférieures sur l'étirement se transmettent des spanneurs aux spanneurs multichemins. Enfin, nous améliorons le premier résultat dans deux cas simples (Théorèmes 2,3) et ensuite en utilisant les résultats des spanneurs tolérants aux fautes (Théorème 4).

Chapitre 4 : Vertex-disjoint multipath spanners. Ce chapitre fait suite au chapitre sur les spanneurs multichemins arêtes disjoints. Il est proposé de s'intéresser ici à des p -chemins sommets disjoints. Après une définition formelle de la métrique multichemins sommets disjoints (Définition 7), et la remarque (Remarque 1) que cette métrique ne vérifie pas la propriété de l'inégalité triangulaire, nous introduisons la notion de spanneur multichemins sommets disjoints (Définition 8).

Sont ensuite exposés les résultats sur les spanneurs multichemins sommets disjoints obtenus au cours de la thèse. D'abord, nous remarquons que bien que les spanneurs tolérants aux fautes soient liés aux spanneurs multichemins, un spanneur tolérant $p - 1$ fautes peut avoir un étirement non borné pour la métrique multichemins sommets disjoints (Remarque 2). Ensuite, nous démontrons que dans certains cas particuliers de spanneurs tolérants aux fautes, il est possible d'avoir un étirement borné (Théorème 5). Ensuite, nous montrons qu'en augmentant le nombre de fautes supportées par un spanneur tolérant aux fautes, il est possible d'améliorer considérablement l'étirement quand celui-ci est considéré en tant que spanneur multichemins sommets disjoints (Théorèmes 7,8). Enfin, nous montrons avec le Théorème 10

qu'un algorithme ad-hoc permet d'obtenir un spanneur multichemins sommets disjoints avec un étirement additif dont la composante multiplicative est inférieure à celles obtenues dans les résultats antérieurs.

Chapitre 5 : Conclusions & perspectives. Ce chapitre traite des questions non abordées au cours de cette thèse, et aborde les orientations futures à donner à la recherche sur les spanneurs multichemins.

Chapter 1

Presentation

This document is about the study of multipath spanners, as an extension of classical graph spanners.

A spanner H of a graph G is a spanning subgraph (ie., $V(H) = V(G)$) such that for any pair of graph vertices $a, b \in V(G)$, the distance (measured as the length of a shortest path from a to b) measured using edges from H $d_H(a, b)$ isn't too much stretched compared to the distance in the original graph $d_G(a, b)$. Indeed there exists a stretch factor (α, β) such that for any $a, b \in V(G)$, $d_H(a, b) \leq \alpha \cdot d_G(a, b) + \beta$. A spanner such that $\beta = 0$ is called a multiplicative spanner.

Spanners were initially introduced by Peleg and Schäffer [PS89] for unweighted graphs, then extended to weighted graphs in [ADD⁺93]. Spanners have been the subject of many publications. A survey was made by Pettie in [Pet07].

Motivated by multipath routing and after the remark that the concept of spanner seems relevant for any non-decreasing graph metric—a non decreasing graph metric is such that any removal of edges will increase the distance for any pair of vertices—we introduce the notion of multipath spanner. After an introduction to the topic, we present our results concerning edge-disjoint multipath spanners in Chapter 3 and the results concerning vertex-disjoint multipath spanners in Chapter 4.

This document is organized as follows, each chapter being largely independant of the preceding ones:

Chapter 2: Introduction. In this chapter we first remind the reader to basic concepts of graph theory needed in the rest of the document, as well as some results about spanners.

Multipath routing being strongly linked to fault-tolerance, we present some results about fault-tolerant spanners.

We then present the extension proposed in this thesis: multipath spanners. To this effect, we first begin by defining the p -multipath metric which relies on p -paths, then we define multipath spanners using this metric.

Finally, we present the theorems obtained during this thesis. First those concerning of edge-disjoint spanners, then in those concerning vertex-disjoint multipath spanners.

Chapter 3: Edge-disjoint multipath spanners. In this chapter, after reminding the reader to the edge-disjoint multipath graph metric, edge-disjoint p -multipath spanners are defined i

We then present the results obtained concerning edge-disjoint multipath spanners: first we note in Theorem 1 that a trivial algorithm using classical multiplicative spanners from [ADD⁺93] can yield edge-disjoint p -multipath spanners with a stretch multiplied by p compared to the original stretch. Then we show with Proposition 2 that lower bounds on stretch can be translated from spanners to multipath spanners. Finally the first result is improved. Firstly in two particular cases with Theorems 2,3 and secondly using results from the theory of edge-fault-tolerant spanners with Theorem 4.

Chapter 4: Vertex-disjoint multipath spanners. In this chapter it is proposed to examine vertex-disjoint p -paths instead of edge-disjoint p -paths. After the definition of the vertex-disjoint multipath graph metric, and the remark that it does not satisfy the triangle inequality, we introduce the notion of vertex-disjoint multipath spanner (Definition 8).

Next we present the results obtained on vertex-disjoint multipath spanners. First we note that although vertex multipath spanners and fault-tolerant spanners are linked, it is generally false that a vertex fault-tolerant spanner is a vertex-disjoint multipath spanner (Remark 2). Despite this negative result, we exhibit a property in Theorem 5 that ensures that $(p - 1)$ -vertex fault-tolerant spanners are also vertex-disjoint p -multipath spanners with a bounded stretch. Then we show in Theorems 7 and 8 that by increasing the number of faults tolerated by a fault-tolerant spanner it is possible to get a vertex-disjoint multipath spanner with a better stretch. Finally, we show in Theorem 10 that an ad-hoc algorithm can yield a vertex-disjoint multipath spanner with a stretch whose multiplicative constant is inferior to those of previous results.

Chapter 5 : Conclusions & perspectives. This chapter is concerned with the topics left untreated during this thesis, and shows the future works which should be carried concerning multipath spanners. In particular we raise the

problem of finding which non-decreasing graph metrics can sustain a spanner trade-off.

Chapter 2

Introduction

2.1 Topic introduction

This document is about spanners. Spanners are combinatorial objects which appear in multiple fields of computer science and information theory: graph theory, computational geometry, networking, computational biology, distributed computing etc.. Spanners are graphs and as such belong to the field of graph theory.

2.1.1 Graphs and distance

An unweighted graph, denoted by $G = (V, E)$ is the association of two sets, V and E :

- $V = V(G)$ is called the set of vertices. Its size is denoted by $n = |V|$.
- $E = E(G)$ is called the set of edges. An edge $e \in E$ is simply an unordered pair of vertices $e = (u, v); u, v \in V$. Its size is denoted by $m = |E|$.

A weighted graph is a graph where each edge e has an associated weight—generally a non-negative real—denoted by $\omega(e)$. This allows to model the cost of using the edge, when going from one vertex to another. Another frequent extension are directed graphs, where the edges are now ordered pairs. This helps to model cases where an edge can only be crossed in one direction. An undirected graph can generally be modeled as a directed graph by replacing every undirected edge by two directed edges going in opposite directions, or simply, by replacing each unordered pair by the two ordered pairs composed of the two vertices.

Graphs allow to model varied problems. We are specially interested in network problems. To this end, graph vertices represent units of computations

(i.e., computers) while edges represent communication links between these computers. This type of model is adapted to the study of large communication networks like the Internet as well as the study of smaller sensor networks.

A parameter often taken into account when dealing with graph problems is the distance. The distance between two vertices u, v of G (i.e., $u, v \in V(G)$), denoted by $d_G(u, v)$, is the minimum cost of a path connecting u to v using edges of G , counting weights.

Many network problems can be modeled with graphs. A first example is the node synchronization problem. In the context of distributed computing it is necessary that every unit wakes up at the same moment. To obtain synchronization, messages are exchanged using the links, from neighbour to neighbour. It is also useful to limit the number of such links in order to have the best efficiency, as shown by the work of Peleg and Ullman in [PU89].

Another frequent problem in communication networks is to send a message from one vertex to another while minimizing the distance the message has to travel—hence the delay. If one wants to answer without taking too much memory one solution is to spare some links while guaranteeing that the graph is still connected. Indeed, if some vertices were at finite distance before removing links they must still be at finite distance afterwards.

2.1.2 Spanners

Spanners are graph skeletons. A *spanner* of a graph $G = (V, E)$ is a spanning subgraph $H = (V, E')$, $E' \subset E$ (i.e., a subgraph such that every vertex from the original graph belongs to the subgraph) such that the distance in H between every pair of vertices is controlled. A good spanner H will be such that the distance measured using its edges isn't too stretched compared to the original graph. More precisely, there exists a stretch factor between G and H . It is a couple of reals (α, β) such that the distance between any pair of vertices u, v measured in H and denoted by $d_H(u, v)$ is no more than α times the distance measured in G plus β :

$$d_H(a, b) \leq \alpha \cdot d_G(a, b) + \beta$$

H is then called an (α, β) -spanner of G . When $\beta = 0$ H is said to be a *multiplicative spanner*. In that case the spanner will be simply called an α -spanner. On the contrary, an *additive spanner* is a spanner in which $\alpha = 1$.

Being sparse is one of the principal reasons for which we study spanners. For a given stretch, the best spanner will be the one with the least number of edges. Conversely, for a given number of edges, the best spanner will be the one with the smallest stretch. More precisely, if $\alpha' < \alpha$ or $\alpha = \alpha'$ & $\beta' < \beta$ then the stretch (α', β') is considered better than (α, β) .

Among classical results, we can cite the whole family of multiplicative spanners, obtained by Althöfer et al. in [ADD⁺93]:

Theorem ([ADD⁺93]) *For any weighted graph G and any strictly positive integer k , one can build in polynomial time a spanner H of G with at most $O(n^{1+1/k})$ edges and of stretch $2k - 1$.*

Another classical result is the additive $(1, 2)$ -spanner theorem, by Aingworth et al.:

Theorem ([ACIM99]) *For any weighted graph G one can build in polynomial time a spanner H of G with at most $O(n^{3/2})$ edges and stretch $(1, 2)$.*

The family of additive stretch spanners is much more restricted in the sense that it is an open question to know whether for every $k > 0$, a stretch $(1, f(k))$ spanner exists with $O(n^{1+1/k})$ edges. However, recently a $(1, 6)$ -spanner was obtained by Baswana et al. :

Theorem ([BKMP10]) *For any unweighted graph G one can build in polynomial time a spanner H of G with $O(n^{4/3})$ edges and stretch $(1, 6)$.*

Number of edges vs. stretch trade-off. We define the girth of a graph G as the length of a smallest cycle of G . The trade-off between the number of edges and the stretch is related to a conjecture by Erdős:

Conjecture ([Erd64, ES82]) *For any integer $k > 0$ and any integer n , there exists a graph with n vertices, $\Omega(n^{1+1/k})$ edges and girth at least $2k + 2$.*

On such a graph G , the removal of any edge $e = u - v$ increases the distance $d_G(u, v)$ from 1 to $d_{G \setminus e}(u, v) \geq 2k + 1$ —because at best the edge belongs to a cycle of length $2k + 2$. See Figure 2.1. This implies that every spanner algorithm which yields multiplicative spanners with stretch strictly less than $2k + 1$ will not remove any edge from this graph. As a consequence, if Erdős's conjecture is true, then a spanner algorithm yielding (α, β) -spanners with $\alpha + \beta \leq 2k$ will return on some instances spanners with $\Omega(n^{1+1/k})$ edges¹.

This conjecture was proved in [Wen91] for $k = 1, 2, 3$ and 5.

¹ $\Omega()$ is the domination symbol: $f(n) = \Omega(g(n))$ when there exists k such that for any sufficiently large n , $f(n) \geq k \cdot g(n)$.

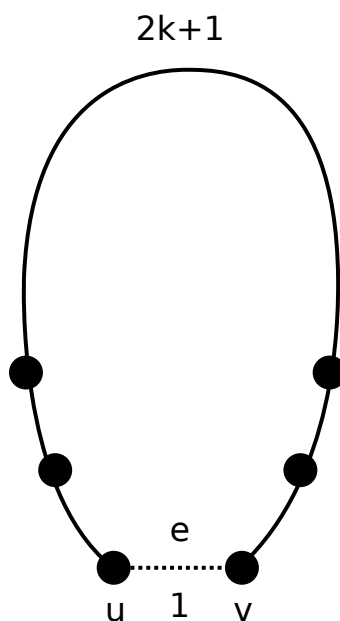


Figure 2.1: Edge e removal increases the distance $d_G(u, v)$ from 1 to $2k + 1$.

2.1.3 Spanner extensions

2.1.3.1 Fault tolerant spanners

Graphs used as a model of real networks are naturally set in a dynamic setting: vertices or edges appear and disappear all the time. In such a context it is desirable to have spanner algorithms which take into account the dynamic aspect of the network. One possibility is to compute updates of the spanner each time a change appears in the network. In that direction Baswana and Sarkar conducted a study in [BS09] where the authors present a spanner in which the update time is polylogarithmic and can be calculated in a distributed manner.

However, for certain topologies it might be desirable to avoid a new computation each time the graph changes. Indeed, if the graph is a modelisation of a communication network as the Internet, the links or vertices suffer mostly from transient faults without impacting the rest of the graph. This kind of behaviour is observed in transportation networks, where faults are mostly the results of roadworks or accidents. Overall, major modifications are slow enough to allow for a complete update. In these conditions, it is more sensible to consider a structure avoiding a new computation in the case of a small number of failures.

To this effect, the concept of fault-tolerant spanner for a general graph was

introduced by Chechik et al. in [CLPR10]:

Definition ([CLPR10]) *A vertex (respectively edge) f -fault tolerant (α, β) -spanner H of a graph G is a spanning subgraph of G such that for every vertex pair a, b and every fault set $F \subset V(G) \setminus \{a, b\}$ (respectively $F \subset E(G)$) with $|F| \leq f$ we have:*

$$d_{H \setminus F}(a, b) \leq \alpha \cdot d_{G \setminus F}(a, b) + \beta$$

where $H \setminus F$ stands for the graph $(V(H) \setminus F, E(H))$ if F is a vertex set, and for the graph $(V(H), E(H) \setminus F)$ if F is an edge set.

Initially developed in the context of geometric graphs [LNS98, CZ04], Chechik et al. [CLPR10] extended the notion of fault-tolerant spanner to general graphs. This article studies both edge fault-tolerant spanners and vertex fault-tolerant. More specifically the authors construct an f -vertex-fault tolerant $(2k - 1)$ -spanner with $O(f^2 k^{f+1} \cdot n^{1+1/k} \log^{1-1/k} n)$ edges. In the edge-fault setting, the authors obtain an f -edge-fault tolerant $(2k - 1)$ -spanner with $O(f \cdot n^{1+1/k})$ edges. If constructions with high probability are allowed, then the works from Dinitz and Krauthgamer [DK11] improved these results, with a generic procedure taking any standard multiplicative spanner construction as a parameter and returning an f fault-tolerant spanner. In particular, when used with the greedy algorithm the authors obtain f -vertex-fault tolerant spanners with $\tilde{O}(f^2 n^{1+1/k})$ edges².

2.1.3.2 Non-decreasing graph metrics

Another—independent— way to extend spanners is to remark that the notion of spanner is oblivious of the specific graph metric. Let δ be a non-decreasing graph metric, that is a symmetric positive function defined on pairs of vertices and which has a higher value when computed on a subgraph. More precisely δ is such that for every graph G , every subgraph H and every pair of vertices $u, v \in V(H)$, $\delta_G(u, v) \leq \delta_H(u, v)$.

We then define spanners on this graph metric:

Definition *A subgraph H of a graph G is an (α, β) -spanner of G with respect to the graph metric δ if:*

$$\forall u, v \in V(G); \delta_H(u, v) \leq \alpha \cdot \delta_G(u, v) + \beta$$

As for standard spanner, we will say that H is an α -spanner of G for the graph metric δ if it is an $(\alpha, 0)$ -spanner of G for the graph metric δ .

² $f(n) = \tilde{O}(g(n))$ when $f(n) = O(g(n) \cdot \log^k(g(n)))$ for some k

2.1.3.3 Size-stretch trade-off

The extensions seen above are *a priori* applicable to any graph metric. However, not every graph metric allows for the creation of spanners with good size-stretch trade-offs.

For instance, in the context of directed graphs Thorup and Zwick show in [TZ05] that there are no spanners of a general directed graph with $o(n^2)$ edges and constant stretch for the one way distance. The one way distance between two vertices u and v is defined as the minimum cost of a path going from u to v using only forward edges. Nevertheless, as shown by Roditty et al. in [RTZ08] a size-stretch trade-off exists for the round-trip distance, defined as the sum of a minimum cost of a directed path from u to v , and a minimum cost directed path from v to u . For instance, they show that a general directed graph has a $(2k + \epsilon)$ -spanner for the roundtrip distance with $O(\min\{(k^2/\epsilon)n^{1+1/k}\log(nW), (k/\epsilon)^2n^{1+1/k}(\log n)^{2-1/k}\})$ edges, where W is the maximum edge weight. They also show that their scheme allow for compact roundtrip routing, improving the previous results from Cowen and Wagner in [CW00].

A natural question raised by these extended spanners is to know for which graph metrics there exists such a size-stretch trade-off.

In the rest of this document, we are interested in the multipath graph metric, and in spanners built upon this graph metric. Roughly speaking, the multipath graph metric assigns to each pair of vertices u, v of G the minimum sum of the costs of a fixed number of disjoint paths between u and v . We then try to build spanner according to this graph metric.

2.2 Multipath spanners

2.2.1 Motivation

Our interest in multipath spanners stems from the need of multipath routing in networks. Using multiple paths between a pair of vertices is an obvious way to aggregate bandwidth. Additionally a classical approach to quickly overcome failures consists in pre-computing fail-over paths which are disjoint from primary paths [KKKM07, PSA05, NCD01]. Multipath routing has been extensively studied in ad-hoc networks for load balancing, fault-tolerance, higher aggregate bandwidth, diversity coding, minimizing energy consumption (see [MTG03] for a quick overview). Heuristics have been proposed to provide disjoint routes [NCD01, LG01] in on-demand protocols. There is a wide variety of optimization requirements when using several paths

between pairs of nodes. However, using edge-disjoint or vertex-disjoint paths is a recurrent concern in optimizing routing in networks. Using disjoint paths is a subject of study in itself [Kle96] and has many problem variants.

Considering only a subset of links is a practical concern in link state routing in ad-hoc networks [JV09]. This raises the problem of computing spanners for the multipath graph metric. Additionally, spanners are a key ingredient in the design of compact routing schemes [PU89, TZ05]. Designing multipath spanners is thus a first step toward multipath compact routing.

2.2.2 Multipath distance

To define multipath spanners, we define the multipath distance on which they are built. We define two versions of this multipath distance: the *edge-disjoint* multipath distance and the *vertex-disjoint* multipath distance. This will in turn define the two types of multipath spanners which are studied in the rest of this document.

The multipath distance is defined with the help of multipaths—or p -paths—in the same way the standard graph distance is defined with the help of paths. A p -path \mathcal{Q} from u to v is a subgraph composed of p (different) paths $Q_1 \dots Q_p$ going from u to v , a path being simply a subgraph composed of an uninterrupted chain of edges going from vertex u to vertex v :

$$\mathcal{Q} = Q_1 \cup Q_2 \cup \dots \cup Q_p$$

where $A \cup B$ in the context of subgraphs means the graph $(V(A) \cup V(B), E(A) \cup E(B))$.

Note that 1-paths are exactly classical paths. We show an example of a 4-path in Figure 2.2.

An edge-disjoint p -path will additionally requires that paths of \mathcal{Q} are pairwise edge-disjoint:

$$\forall 1 \leq i < j \leq p, E(Q_i) \cap E(Q_j) = \emptyset$$

Note that Figure 2.2 is not a 3-path despite containing 3 edge-disjoint paths. Indeed it cannot be obtained as an union of three edge-disjoint paths. However it contains a 3-path, as shown on Figure 2.3.

Similarly, a vertex-disjoint p -path is such that paths composing \mathcal{Q} are pairwise vertex-disjoint except for the two extremities u and v :

$$\forall 1 \leq i < j \leq p, V(Q_i) \cap V(Q_j) = \{u, v\}$$

Note that as a consequence of the definition, a vertex-disjoint p -path is also edge-disjoint.

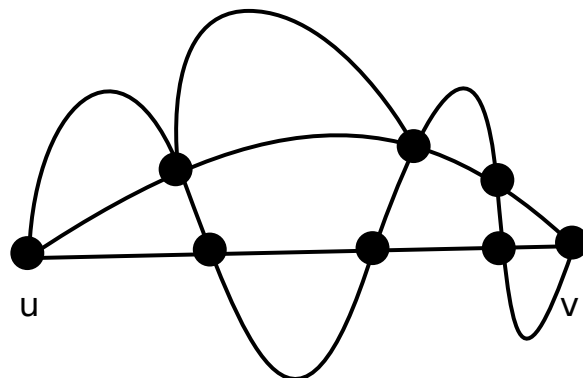


Figure 2.2: This subgraph can be obtain as a combination of four paths, and hence is a 4-path.

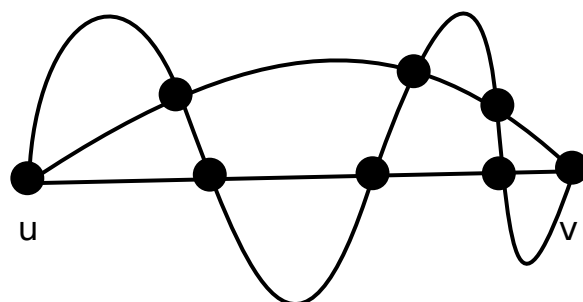


Figure 2.3: The 4-path shown in Figure 2.2 contains only three edge-disjoint paths, but is not in itself a 3-path. Is shown here one of the 3-path contained in the original 4-path.

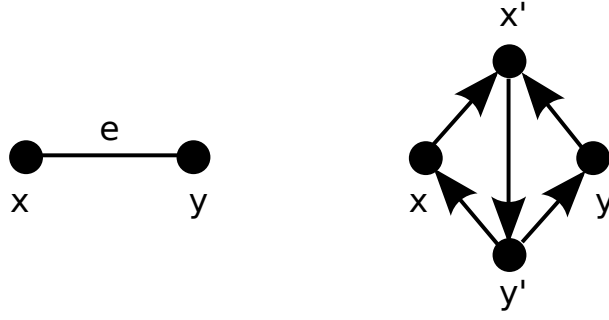


Figure 2.4: The process into which an edge $e = x - y$ is replaced by a directed path, ensuring it is crossed only once. For the flow procedure only the edge $x' \rightarrow y'$ has capacity 1 and weight $\omega(e)$. The other are set to 0.

As with any other subgraph, the cost of a p -path will simply be the sum of its edges' weights:

$$\omega(\mathcal{Q}) = \sum_{e \in E(\mathcal{Q})} \omega(e)$$

We then define the two different versions of the p -multipath distance: the *edge-disjoint p -multipath distance* and the *vertex-disjoint p -multipath distance*. They are defined as the minimum cost of an edge-disjoint (respectively vertex-disjoint) p -path:

$$d_G^p(u, v) = \inf_{\mathcal{Q}} \{ \omega(\mathcal{Q}) : \text{edge-disjoint } p\text{-path } \mathcal{Q} \text{ from } u \text{ to } v \}$$

and

$$v d_G^p(u, v) = \inf_{\mathcal{Q}} \{ \omega(\mathcal{Q}) : \text{vertex-disjoint } p\text{-path } \mathcal{Q} \text{ from } u \text{ to } v \}$$

If there are less than p paths between u and v , then the distance is set to $+\infty$.

The edge-disjoint p -multipath distance $d_G^p(u, v)$ between any pair of vertices u and v is computable in polynomial time: first replace every edge $e = x - y$ of the graph by the directed path $x \rightarrow x' \rightarrow y' \rightarrow y \rightarrow x' \rightarrow y' \rightarrow x$, as shown on Figure 2.4 and assign null weight and infinite capacity for every edge except those of the type $x' \rightarrow y'$ for which the capacity must be set to 1 and the weight to $\omega(e)$. Then apply any minimum-cost flow algorithm using u as the source and v as the sink, stopping the search when reaching p . As the capacities are integral the search should not exceed p steps. The final distance is the sum of the weights obtained so far (see the chapter “Minimum Cost Flows”, pages 129–133 of [Law76] for example).

To compute the vertex-disjoint multipath distance $v d_G^p(u, v)$, one can apply the same transformation to the graph as shown in Chapter 4: Figure 2.5 shows the procedure: every undirected edge $e = x - y$ is replaced by two directed

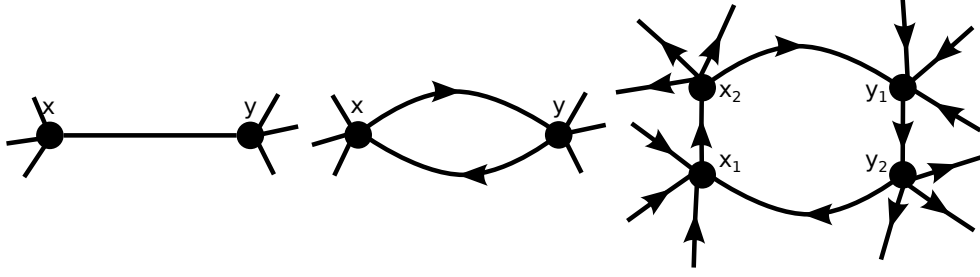


Figure 2.5: Process by which an undirected graph is transformed into a suitable directed graph to compute the vertex-disjoint p -multipath distance d^p_v .

edges going in opposite direction, $x \rightarrow y$ and $x \leftarrow y$, both of weight $\omega(e)$ and unit capacity. Then every vertex x is split into two, x_1 and x_2 , every inward edge into x is made to arrive to x_1 and every outward edge leaving x is made to leave x_2 instead. Then a supplementary edge of null cost and infinite capacity is added from x_1 to x_2 . This ensures that edge-disjoint paths in this new graph are vertex-disjoint in the original graph. We can then apply any minimum cost maximum flow algorithm to this graph to find p paths between u_2 and v_1 , which are easily transformed back into the original graph, allowing the total cost to be computed.

2.2.3 Multipath spanners

As suggested in section 2.1.3.2, we plug the two graph metrics d^p_e and d^p_v into the definition of spanner to create the two different versions of multipath spanners.

We first define the edge-disjoint multipath spanners:

Definition An edge-disjoint p -multipath (α, β) -spanner H of G is a subgraph such that $\forall u, v \in V$, $d^p_H(u, v) \leq \alpha \cdot d^p_G(u, v) + \beta$.

Next we define vertex-disjoint multipath spanners:

Definition A vertex-disjoint p -multipath (α, β) -spanner H of G is a subgraph such that $\forall u, v \in V$, $d^p_H(u, v) \leq \alpha \cdot d^p_G(u, v) + \beta$.

These two definitions revert to the classical spanners when 1-paths are considered, that is when $p = 1$.

In the rest of this document we study the properties of multipath spanners and show how to build them.

2.3 Our results

2.3.1 Edge-disjoint multipath spanners

Chapter 3 deals with the study of edge-disjoint multipath spanners. We present the different results obtained in relationship with edge-disjoint spanners.

We first prove in Theorem 1 that we can use standard multiplicative spanner algorithms to build for any weighted graph an edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanner with $O(p \cdot n^{1+1/k})$ edges.

We then show in Proposition 2 that a simple argument allows us to translate known lower bounds on the number of edges of a spanner to multipath spanners. We then further refine in Section 3.2.3 the proof of the first theorem to improve the stretch in the special case of 2-paths and using 3-spanners.

Section 3.2.4 shows that we can recycle the ideas from the $(1, 2)$ -spanner algorithm to create an edge-disjoint 2-multipath $(2, 8W)$ -spanner³ with roughly the same number of edges. We provide an example to show that the analysis of our algorithm is tight with respect to the stretch.

Finally, we show in Section 3.2.5 that there exists a relationship between edge fault-tolerant spanners and edge-disjoint multipath spanners, which allows us to obtain edge-disjoint p -multipath spanners with a much better stretch than what we obtained in Theorem 1 at the cost of an increased number of edges.

2.3.2 Vertex-disjoint multipath spanners

In Chapter 4 we investigate vertex-disjoint spanners. First we note that the vertex-disjoint p -multipath graph metric—unlike the edge-disjoint graph metric—is not a proper graph distance. We then give the definition of vertex-disjoint p -multipath spanners and present the results obtained about vertex-disjoint p -multipath spanners.

We first show that although vertex fault-tolerant spanners are related to vertex-disjoint multipath spanners, there exist fault tolerant spanners with an unbounded multipath stretch.

In spite of this negative result, we show (Section 4.2.2) that it is possible to overcome this difficulty provided the fault-tolerant spanners have an additional property. More precisely, we prove that there exist $(p - 1)$ -fault tolerant spanners which are also p -multipath spanners with a large stretch independent of the number of edges of the graph. We therefore provide a way to use existing fault-tolerant spanners algorithms to get p -multipath spanners.

³ W is used to note the graph's maximum edge weight.

Moreover we show that these spanners can be constructed efficiently by a distributed algorithm.

While the previous analysis was simple, the obtained stretch is too large to be of any use. We therefore present in Section 4.2.3 a different analysis. We show that fault-tolerant spanners tolerating more faults than p can be vertex-disjoint p -multipath spanners with the same stretch, thereby improving largely on the stretch from the previous section while paying an increase in the number of edges.

Finally, we show in Section 4.2.4 that despite the vertex-disjoint multipath graph metric not being a proper distance and in the case of only 2-paths, we can design an ad-hoc algorithm inspired by the algorithm from Section 3.2.4 which can yield vertex-disjoint 2-multipath $(2, O(W))$ -spanners.

Chapter 3

Edge-disjoint multipath spanners

In this chapter, we consider the case of *edge-disjoint multipath spanners*, as an extension of spanners, defined in [PS89] for general graphs.

A spanner H of a graph G is a spanning subgraph such that for any pair of vertices the distance measured as the length of a shortest path using only edges of H is no more than some factor α than the distance measured in the original graph G : $d_H(u, v) \leq \alpha \cdot d_G(u, v)$.

We note that this definition could be in principle extended to any graph metric such that the measure of distance between two nodes is larger in every subgraph. Therefore we introduce edge-disjoint multipath spanners, in the hopes that they will help to solve the problems related to multipath routing.

To this effect, we introduce the edge-disjoint multipath graph metric (Section 3.1.1). A multipath spanner will simply be a spanner defined on this distance.

3.1 Preliminaries

We consider a (possibly weighted) undirected graph $G = (V, E)$ with weight function ω . Let $G \rightarrow V(G) = V$ be the function which associates to a graph its set of vertices, and $G \rightarrow E(G) = E$ be the function which associates the set of edges. We set $n = |V|$.

The *cost* of any subgraph H of G is the sum of the weights of its edges. It is denoted by $\omega(H) = \sum_{e \in E(H)} \omega(e)$. We set $\omega(H) = 0$ if $E(H) = \emptyset$.

3.1.1 Multipath distance

Definition 1 A p -path $\mathcal{Q} = \{Q_1, \dots, Q_p\}$ from a vertex u to a vertex v is a subgraph of G composed of p edge-disjoint paths from u to v , i.e., $E(Q_i) \cap E(Q_j) = \emptyset, \forall 1 \leq i <$

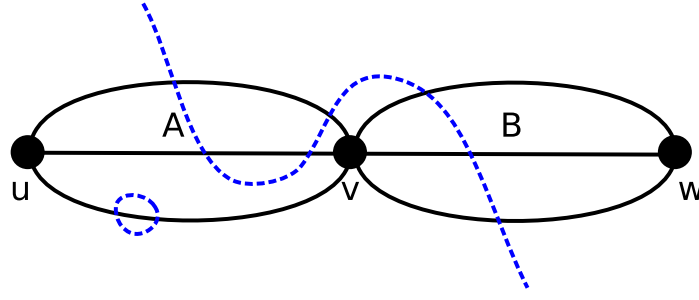


Figure 3.1: Any cut separating u from w has capacity at least 3

$j \leq p$.

Definition 2 The p -multipath distance between two vertices u and v , denoted by $\mathring{d}_G^p(u, v)$, is the minimum cost of a p -path from u to v . We set $\mathring{d}_G^p(u, v) = \infty$ if there are no p edge-disjoint paths from u to v . The p -multipath distance in a subgraph H is denoted by \mathring{d}_H^p .

When the graph is omitted it is implied to be G .

In the following, we prove that \mathring{d}^p is a metric endowed with the triangle inequality. This property is essential to our needs to prove the main results of this chapter.

Proposition 1 \mathring{d}^p is a metric endowed with the triangle inequality.

Proof. \mathring{d}^p clearly satisfies $\mathring{d}^p(u, v) = 0$ if and only if $u = v$. The symmetry is also obvious due to the fact that we deal only with undirected graphs. We prove next the triangle inequality. Let u, v, w be three vertices, A a p -path of cost $\mathring{d}_G^p(u, v)$ between u and v , and B a p -path of cost $\mathring{d}_G^p(v, w)$ between v and w . We show that the subgraph $H = (V(A) \cup V(B), E(A) \cup E(B))$ contains a p -path between u, w .

Let (U, W) ($U, W \subset V(H)$) be a cut (a cut is simply a partition of vertices) of capacity t (the capacity of the cut is the number of edges which have an endpoint in U and the other in W) which separates u from w , i.e. $u \in U$ and $w \in W$ as shown on Figure 3.1. Since the cut separates u from w , v is either in U or W . Suppose it is in U . The cut effectively separates v from w . Since $\mathring{d}_G^p(v, w)$ is finite we know that there exists at least p paths from v to w in B , and so by the min-cut max-flow theorem our cut has capacity at least p , hence $t \geq p$. The same is true when v is in W . Since any cut separating u from w has at least capacity p , we know by the min-cut max-flow theorem that in H there are at least p paths between u and w . Hence $\mathring{d}^p(u, w) \leq \mathring{d}^p(u, v) + \mathring{d}^p(v, w)$.

□

3.1.2 Multipath edge-disjoint spanners

We define an *edge-disjoint p -multipath spanner* as a spanner for the edge-disjoint p -multipath distance.

This definition is in effect the transposition of the classical spanner definition from [PS89] to the new metric.

Definition 3 Let $G = (V, E)$. Then $H = (V, E' \subset E)$ is an *edge-disjoint p -multipath (α, β) -spanner* of G if $\forall u, v \in V$, $d_H^p(u, v) \leq \alpha \cdot d_G^p(u, v) + \beta$.

When $\beta = 0$, H is simply called an α -spanner.

This definition is identical to the traditional definition of spanners for $p = 1$.

3.2 Results

In this part, we prove the following results:

- Theorem 1: every weighted graph admits an edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanner with $O(p \cdot n^{1+1/k})$ edges.
- Proposition 2: lower bounds for one-path spanners are also valid for edge-disjoint p -multipath spanners.
- Theorem 2: every unweighted graph admits an edge-disjoint 2-multipath 3-spanner with $O(n^{3/2})$ edges.
- Theorem 3: every multi-edge weighted graph with n vertices and largest edge-weight W has an edge-disjoint 2-multipath $(2, 8W)$ -spanner with $O(n^{3/2})$ edges.
- Theorem 4: a $O(kp^2 \cdot \log(W))$ -fault tolerant $(2k - 1)$ -spanner of a weighted graph G is an edge-disjoint p -multipath $(2k - 1)$ -spanner of G .

The first four results were published in coöperation with Cyrille Gavoille and Laurent Viennot in [GGV10].

Theorem 4 was published in coöperation with Shiri Chechik and David Peleg in [CGP12].

3.2.1 Edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanner

In this section we show that a simple algorithm can produce a multipath spanner.

Theorem 1 For all integers $k, p \geq 1$, every multi-edge weighted graph with n vertices has an edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanner with at most $p \cdot n^{1+1/k}$ edges.

To prove this theorem, we make use of what we call *iterative spanners*.

Definition 4 A p -iterative s -spanner of G is a subgraph $H = \bigcup_{i=1}^p H_i$, where H_i is any 1-multipath s -spanner of $G \setminus \bigcup_{j < i} H_j$.

We observe that the union of p such 1-multipath spanners is actually a p -multipath spanner.

Proof. Let $H = \bigcup_{i=1}^p H_i$ be a p -iterative $(2k - 1)$ -spanner of G , where H_i is a $(2k - 1)$ -spanner of G with less than $n^{1+1/k}$ edges. Each H_i can be constructed (cf. [ADD⁺93]). Hence, H has less than $p \cdot n^{1+1/k}$ edges.

We now prove that H is an edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanner. Let u, v be two vertices of G . If there is no p -path from u to v in G , then $d_G^p(u, v) = \infty$. In particular, $d_H^p(u, v) \leq p \cdot (2k - 1) \cdot d_G^p(u, v)$. So, we assume there exists a p -path from u to v . Let P be any minimum-cost p -path from u to v in G . We have $\omega(P) = d_G^p(u, v)$.

For an edge $e \notin H$, we denote by $H_i(e)$ the simple path which replaces the edge e of G in the i -th spanner member of H . Observe that, for each i , $\omega(H_i(e)) \leq (2k - 1) \cdot \omega(e)$ because $e \in G \setminus H_i$ and H_i has stretch $2k - 1$.

Given P and H , we define the subgraph F as follows:

$$F := (P \cap H) \cup \bigcup_{e \in E(P \setminus H)} \bigcup_{i=1}^p H_i(e).$$

Clearly, $F \subseteq H$, since each edge $e \in P$ is either in H or is replaced by $H_i(e)$ for some i . Moreover, we have $\omega(F) \leq p \cdot (2k - 1) \cdot \omega(P)$ because:

$$\begin{aligned} \omega(F) &\leq \omega(P \cap H) + \sum_{e \in P \setminus H} \sum_{i=1}^p \omega(H_i(e)) \\ &\leq \sum_{e \in P \cap H} \omega(e) + \sum_{e \in P \setminus H} p \cdot (2k - 1) \cdot \omega(e) \\ &\leq \sum_{e \in P} p \cdot (2k - 1) \cdot \omega(e) = p \cdot (2k - 1) \cdot \omega(P). \end{aligned}$$

Therefore, the stretch of H is at most $p \cdot (2k - 1)$.

We now show that F contains a p -path from u to v , and for that we shall use the min-cut max-flow theorem. Consider a cut (X, \bar{X}) with $u \in X$ and $v \in \bar{X}$. Since P is a p -path from u to v , there is a subset C of the cut of at least p edges of P which have one endpoint in X and the other in \bar{X} . Two cases are possible:

1. Every edge of C belongs to F : the cut has capacity at least p in F .
2. One edge of C does not belong to F : p paths were added in F in replacement for this edge, so the minimum cut has capacity at least p .

Therefore, the minimum cut in F is at least p . By the min-cut max-flow theorem there is p edge-disjoint paths from u to v in F . It follows that F contains a p -path from u to v . This completes the proof. \square

The result of Theorem 1 is a bit disappointing in the sense that the stretch of the resulting p -multipath spanner is p times the stretch of the underlying spanner algorithm ($p \cdot (2k - 1)$ versus $2k - 1$). More precisely, there is a hope to show that each individual sub-path of a p -path gets stretched by a factor $(2k - 1)$, hence yielding a global stretch of $2k - 1$.

In all the rest of this part, we study how this $p \cdot (2k - 1)$ bound on the number of edges can be improved.

3.2.2 Lower bounds for edge-disjoint multipath spanners

In this section (taken from [GGV10]), we prove that the stretch s of an edge-disjoint p -multipath spanner cannot be better than the stretch of a 1-multipath spanner with the same number of edges.

For all integers p, n and real $s > 1$, denote by $m_p(n, s)$ the largest integer such that there exists a multi-edge weighted graph with n vertices for which every edge-disjoint p -multipath spanner of stretch $< s$ requires at least $m_p(n, s)$ edges.

The value of $m_p(n, s)$ provides a lower bound on the sparsity of edge-disjoint p -multipath spanners of stretch $< s$. To illustrate this, consider for instance $p = 1$ and $s = 3$. It is known that $m_1(n, 3) = \Omega(n^2)$, by considering the complete bipartite graph $B = K_{\lfloor n/2 \rfloor, \lceil n/2 \rceil}$. Since all cycles of B have length at least 4, every proper subgraph H contains two vertices x and y which are neighbors in B but at distance at least 3 in the spanner. Thus H is an s -spanner with $s \geq 3$. In other words, every s -spanner of B , with $s < 3$ contains all the edges of B that is $\Omega(n^2)$ edges.

Unfortunately, this argument does not transfer to edge-disjoint p -multipath spanners whenever $p > 1$. Indeed, with the same graph B , we get $d_B^p(x, y) = 1 + 3(p - 1)$. And, if (x, y) is removed from any candidate spanner H , we only get $d_H^p(x, y) = 3p$. Hence, the stretch for H so proved is only $d_H^p(x, y)/d_G^p(x, y) = 1 + O(1/p)$. The argument transfers however if multi-edges are allowed.

Proposition 2 *For all integers $n, p \geq 1$ and real $s > 1$, $m_p(n, s) \geq m_1(n, s)$.*

In particular, under the Erdős-Simonovits [Erd64, ES82] Conjecture¹ which implies $m_1(n, 2k + 1) = \Omega(n^{1+1/k})$ for every integer $k \geq 1$ and proved for $k \in \{1, 2, 3, 5\}$ [Wen91], there is a multi-edge unweighted graph with n vertices for which every edge-disjoint p -multipath spanner with stretch $< 2k + 1$ has $\Omega(n^{1+1/k})$ edges.

Proof. Let G be an n -vertex graph with the minimum number of edges such that every spanner of stretch $< s$ has at least $m_1(n, s)$ edges. Let ω be the weight function of G . Clearly, G has $m_1(n, s)$ edges, since otherwise we could remove an edge of G . Observe also that any path between two neighbors x, y of G that does not use the edge (x, y) has length at least s , since otherwise we could remove it from G . In other words, $d_{G \setminus \{(x, y)\}}(x, y) \geq s \cdot \omega(x, y)$.

Let G_p be the graph obtained from G by adding, for each edge of G , $p - 1$ extra multi-edges with same weight. We have $G_1 = G$, and G_p has $p \cdot m_1(n, s)$ edges. Let H be any p -multipath spanner of G_p with $< m_1(n, s)$ edges. There must exist two vertices x, y adjacent in G_p that are not adjacent in H . We have $d_{G_p}^p(x, y) = p \cdot \omega(x, y)$, and $d_H^p(x, y) \geq p \cdot d_{G \setminus \{(x, y)\}}(x, y) \geq p \cdot s \cdot \omega(x, y)$. We conclude that the p -multipath stretch of H is at least $d_{G_p}^p(x, y) / d_H^p(x, y) \geq s$.

In other words, every edge-disjoint p -multipath spanner H of G_p with stretch $< s$ has $\geq m_1(n, s)$ edges, proving that $m_p(n, s) \geq m_1(n, s)$. \square

It remains open to determine whether the same lower bound of $2k - 1$ on the stretch applies if one restricts to simple graphs only.

3.2.3 An unweighted edge-disjoint 2-multipath 3-spanner

In this section (taken from [GGV10]), we focus on unweighted edge-disjoint 2-multipath 3-spanners. The lower bound of Proposition 2 tells us that $\Theta(n^{3/2})$ is the required size of any edge-disjoint 2-multipath 3-spanner. However, the p -iterative $(2k - 1)$ -spanner given by Theorem 1 (with $p = k = 2$) provides an edge-disjoint 2-multipath spanner of stretch 6 only. In fact a finer analysis shows that the same construction yields a 2-multipath 3-spanner.

Theorem 2 *Every multi-edge unweighted graph with n vertices has an edge-disjoint 2-multipath 3-spanner with less than $2n^{3/2}$ edges that can be constructed as a 2-iterative 3-spanner.*

Proof. It is obvious from the construction that a 2-iterative 3-spanner contains less than $2n^{3/2}$ edges. In the following we show the stretch property.

Let G be an unweighted multi-edge graph. For any subgraph H of G , we denote by $|H| = |E(H)|$. Since G is unweighted, note that $|H|$ is the cost of H ,

¹It states that there are n -vertex graphs with $\Omega(n^{1+1/k})$ edges without cycles of length $\leq 2k$.

i.e., its number of edges. Consider any 2-iterative 3-spanner H constructed from G , and call H_1 and H_2 the two spanners constituting H . Let u, v be two vertices, and P be a minimal 2-path in G which connects u and v . In the following, we use a decomposition of P in two simple edge-disjoint paths P_1 and P_2 . If they share a common vertex w , it is called an *intersection point*. P is said to be simple if P_1 and P_2 are in fact vertex-disjoint, i.e., they do not share any intersection point.

We first show that any minimal 2-path P can be decomposed in a sequence of simple 2-paths. Let $u_0 = u, u_1, \dots, u_t = v$ denote the intersection points as they are ordered on the path P_1 from u to v .

Lemma 1 *The intersection points u_0, \dots, u_t appear in the same order on P_2 .*

Proof. Suppose they are not in the same order on the two paths. Then there exists $i < j$ such that u_j appears before u_i on P_2 . The sequence on P_1 (resp. P_2) looks like : $u_0, \dots, u_i, \dots, u_j, \dots, u_t$ (resp. $u_0, \dots, u_j, \dots, u_i, \dots, u_t$). Let P'_1 and P'_2 be the paths realized by interleaving P_1 and P_2 at the crossing point u_i , P'_1 (resp. P'_2) having the same prefix as P_1 (resp. P_2). More precisely, $P'_1 = P_1[u_0, u_j] \circ P_2[u_j, u_t] = P_1[u_0, u_i] \circ P_1[u_i, u_j] \circ P_2[u_j, u_i] \circ P_2[u_i, u_t]$ and $P'_2 = P_2[u_0, u_j] \circ P_1[u_j, u_t]$. The transformation is shown in Fig. 3.2. The 2-path (P'_1, P'_2) has the same cost as (P_1, P_2) . Since the process introduced an explicit loop in P'_1 it can be discarded, resulting in an improvement on the total cost which we supposed to be minimal. This is a contradiction. \square

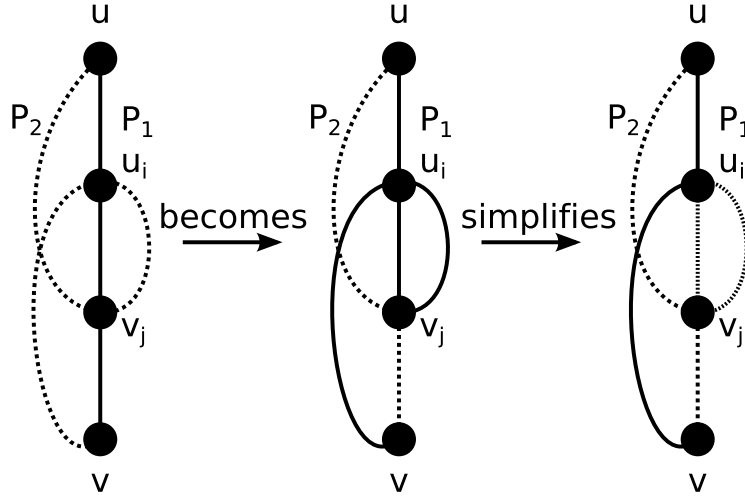


Figure 3.2: Proof of Lemma 1.

Among all 2-paths (P_1, P_2) satisfying $|P_1| + |P_2| = d_G^2(u, v)$, there is at least one which contains a maximum number of intersection points. We now

suppose that $P = (P_1, P_2)$ is such a 2-path and that u_0, \dots, u_t now denote the intersections points of P_1 and P_2 as they are ordered on both paths.

Define P_1^i (resp. P_2^i) as the portion of P_1 (resp. P_2) from u_{i-1} to u_i . Let P^i denote the 2-path formed by the union of P_1^i and P_2^i . Note that by construction P^i is simple. P is indeed the union of P^1, \dots, P^t .

We construct a replacement graph F with edges of H according to the rules bellow. The idea is to replace each edge $x - y \notin H$ of P with a shortest path from x to y in H_1 or H_2 , by using rules which guarantee the stretch property while ensuring the 2-connectivity.

Consider an edge $e \notin H$. Let $H_1(e)$ (resp. $H_2(e)$) be a shortest path in H_1 (resp. H_2) between the end-vertices of e . As H_1 is a 3-spanner, we have $|H_1(e)| \leq 3$. As H_2 is a 3-spanner of $G \setminus H_1$ and $e \notin H_1$, we also have $|H_2(e)| \leq 3$.

We define a subgraph F^i of H by applying the following rules to each 2-path P^i , in the order in which they are presented and for which a schema is shown in Fig. 3.3:

- R0: If P_1^i or P_2^i is a single edge that belongs to $H_j, j \in \{1, 2\}$, add it to F^i . Then for each edge e of the other path: if it belongs to H add it to F^i , if it does not add the replacement path $H_{3-j}(e)$ to F^i .
- R1: If an edge of P^i belongs to H , then add it to F^i .
- R2: For all edges not concerned by R0 nor R1 do:
 - If $e \in P_1^i$ and $H_1(e) \cap P_2^i = \emptyset$, then $F^i = F^i \cup H_1(e)$
 - If $e \in P_2^i$ and $H_2(e) \cap P_1^i = \emptyset$, then $F^i = F^i \cup H_2(e)$
- R3 (disjointedly from R2):
 - If $e \in P_1^i$ and $H_1(e) \cap P_2^i \neq \emptyset$ then $F^i = F^i \cup H_1(e) \cup H_2(e)$.
 - If $e \in P_2^i$ and $H_2(e) \cap P_1^i \neq \emptyset$ then $F^i = F^i \cup H_1(e) \cup H_2(e)$.

F is then the union of all $F^i, i \in \{0, \dots, t\}$.

The following lemmatas (2–5) show that the rules enforce that there is always:

- a 2-path between u and v
- the stretch is controlled.

Lemma 2 F contains a 2-path from u to v .

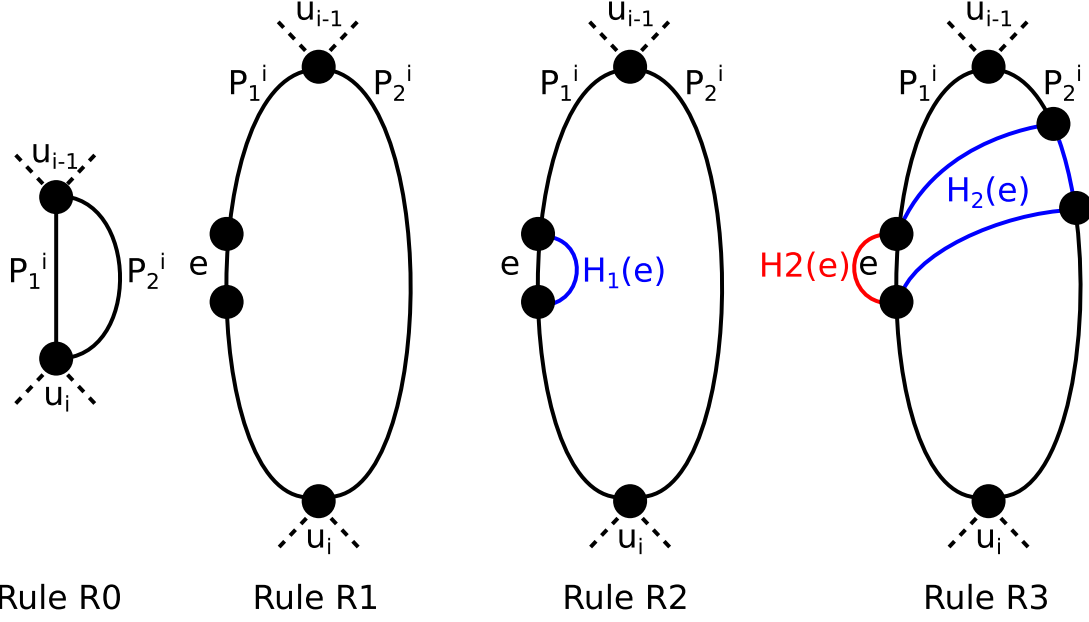


Figure 3.3: The different cases for the rules are shown here for an edge from P_1^i . In blue are shown the edges and paths belonging to H_1 , and in red those from H_2 .

Proof. The proof will show that the capacity of any cut which separates u from v in F is at least two. By the min-cut max-flow theorem it will imply that F contains a 2-path.

Suppose there is a cut $X \subset V$ such that $u \in X, v \in \bar{X}$. Then there exists i such that $u_{i-1} \in X$ and $u_i \in \bar{X}$. Let be e_1 (resp. e_2) an edge in P_1^i (resp. P_2^i) crossing X (i.e., having an extremity in X and the other in \bar{X}).

Several cases are possible:

- R0 has been applied to e_1 or e_2 : then the other edge has a replacement path in F^i , so the cut is at least two in F .
- R1 was applied to both e_1 and e_2 : then each of these edges belong to F^i , are disjoint, so the cut is at least two in F .
- R2 was applied to both e_1 and e_2 : then each of these edges have a disjoint replacement path in F^i , so the cut is at least two in F .
- R1 was applied to one of $\{e_1, e_2\}$, and R2 to the other: then as one of the edges is in F^i and the other has a replacement path in F^i , disjoint from the

other (if it wasn't the case, R3 would have been applied) there are at least two edges from F^i crossing X .

- R3 was applied to either e_1 or e_2 : then there are two disjoint replacement paths for either e_1 or e_2 , which are cut by X , so it is at least two in F .

As the cut is at minimum two in F , by the min-cut max-flow theorem the flow between u and v is at least two in F . \square

Let be $|P^i| = x_0 + x_1 + x_2 + x_3$ the number of edges of P^i , x_j being the number of edges from P^i where rule R_j was applied.

Lemma 3 $|F^i| \leq 3x_0 + x_1 + 3x_2 + 5x_3$.

Proof.

- For rule R0 it is easy to show that the number of edges added in F does not exceed $3x_0$ (one path is a single edge which is added, replacement paths of length at most 3 are added for the edges of the other path).
- For rule R1, the number of edges added in F is exactly x_1 , as these edges are in H .
- For rule R2, the number of edges added is $3x_2$, because the spanners H_1 and H_2 have stretch 3.
- For rule R3, there are at most 5 edges added for each application of the rule because one is already part of F .

\square

Lemma 4 $x_3 \leq x_1$.

Proof. We show that an edge added according to rule R1 can participate in at most one replacement path of an edge considered under rule R3, thus implying $x_3 \leq x_1$. Suppose by contradiction that this is not the case. W.l.o.g. there exists $e_1, e'_1 \in P_1^i, e_2 \in P_2^i$ such that R1 applies to e_2 , and that R3 applies to e_1 and e'_1 . We prove that this may happen only in special cases where $x_3 \leq x_1$ is still satisfied. Let a, b, c, d be the paths such that $a - e_2 - b = H_1(e_1)$ and $c - e_2 - d = H_1(e'_1)$ with $e_2 \in H_1(e) \cap H_1(e'_1)$. (The paths a, b, c, d have length 0, 1 or 2 since H_1 and H_2 are 3-spanners.). There are two possible cases as shown in Fig. 3.4:

1. either a and c are connected together on one endpoint of e_2 — and then b and d are connected to the other end;

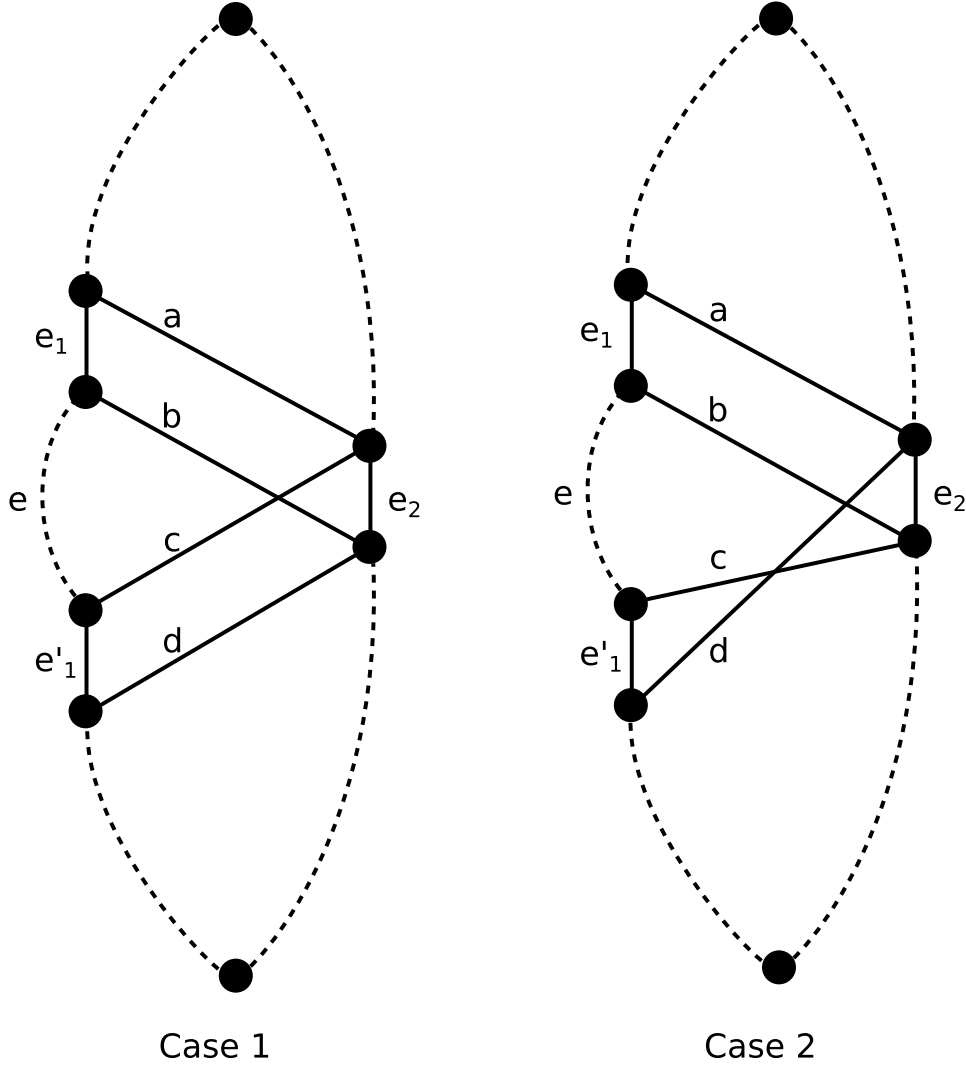


Figure 3.4: Two possible cases.

2. or a and d to one endpoint of e_2 and c and b to the other.

Let e be the sub-path of P_1^i which lies in between e_1 and e'_1 .

In Case 1, we first show $|e| > 0$. Suppose by contradiction that e is empty. Then $a - c$ (resp. $b - d$) is a replacement path for e_1 (resp. e'_1) in H_1 . As $H_1(e_1)$ and $H_1(e'_1)$ are shortest paths, we have $|a| + |c| \geq |a| + 1 + |b|$ and $|b| + |d| \geq |c| + 1 + |d|$, i.e., $|c| > |b|$ and $|b| > |c|$ which is a contradiction.

Now, we show that each of the paths a, b, c, d is composed of a single edge. As we are not in the case of rule $R0$, a and d cannot be both empty. Suppose w.l.o.g. $|a| > 0$. $|H_1(e_1)| \leq 3$ implies $|b| \leq 1$. Indeed, we have $|b| = 1$ as b cannot

be empty since there are no intersection point of P_1 and P_2 in between e_1 and e'_1 . For the same reason, b cannot be composed of an edge of P_2 . As a consequence, d cannot be empty otherwise b would be a shortcut violating the minimality of the cost of P^i . We then similarly show that c is composed of a single edge which is not in P_2 . As $|c| + 1 + |d| \leq 3$ and $|d| > 0$, we obtain $|d| = 1$. Similarly, we have $|a| = 1$.

We now prove that a is in P_2 . If this is not the case, then $a - c$ is a shortcut that can be substituted to $e_1 - e$ in P_1 . This does not increase the cost of the 2-path and it increases the number of intersection points, a contradiction with the choice of P . Similarly, d is in P_2 .

Let us recall what we have obtain so far for Case 1: a and d are edges of P_2 , b and c are edges not in P_2 . P_2^i is thus the path $a - e_2 - d$ and P_1^i is $e_1 - e - e'_1$. Note that $x_1 \geq 3$ since the three edges of P_2^i follow rule R1. We then show that at most one edge of e may fall under rule R3, yielding $x_3 \leq 3 \leq x_1$. Consider an edge e''_1 of e falling under rule R3. We write $e = e' - e''_1 - e''$ where e' and e'' are sub-paths of e . We can write $H_1(e''_1) = b' - e'_2 - c'$ where b' and c' are edges of H_1 and where e'_2 is either an empty path or an edge. As e''_1 follow rule R3, e'_2 must be an edge of P_2^i ($H_1(e''_1)$ must have length 3).

First consider the case where e'_2 is a . Then, e' must be empty (e''_1 must be the first edge after e_1). W.l.o.g. b' is e_1 and c' contains $a \cap e_2$. We must have $|e''| \leq 1$. Otherwise $c' - c$ is a shortcut violating the choice of P . If e'' is an edge, it cannot follow rule R3 as $c' - c$ is a replacement path in H_1 with length 2 and $H_1(e'')$ cannot have length 3. We thus have $x_3 \leq 3 \leq x_1$. The case where e'_2 is d can be treated similarly.

Now consider the case where e'_2 is e_2 . W.l.o.g. b' contains $b \cap e_2$ and c' contains $c \cap e_2$. As $b - b'$ cannot be a shortcut violating the choice of P , we have $|e'| \leq 1$. If e' is an edge, it cannot follow rule R3 as $b - b'$ is a replacement path in H_1 with length 2. Similarly $c - c'$ cannot be a shortcut violating the choice of P . We thus have $|e''| \leq 1$ and if e'' is an edge, it cannot follow rule R3. We again obtain $x_3 \leq 3 \leq x_1$.

In Case 2, we first prove that a, b, c, d are single edges. b and c cannot be empty as there are no intersection point in between e_1 and e'_1 . We thus have $|a| \leq 1$ and $|d| \leq 1$. If a is empty, then d is a shortcut violating the minimality of the cost of P . We thus have $|a| = 1$ which implies $|b| = 1$ as $|H_1(e_1)| \leq 3$. Similarly, we have $|d| = 1$ and $|c| = 1$. Note that b and c cannot be in P_2 as there are no intersection point in between e_1 and e'_1 .

We show that if we are not in Case 1 again, then we have two edges following rule R1 contained in at most two replacement paths of edges following rule R3. As $b - c$ cannot be a shortcut violating the choice of P , we have $|e| \leq 1$. If e is an edge, it admits $b - c$ as a replacement path in H_1 . Thus it

cannot follow rule R3. The path $a - d$ cannot be a shortcut violating the choice of P . This implies that a or d is in P_2 . W.l.o.g., suppose that a is in P_2 ($e_1 \cap a$ is indeed u_{i-1}). d cannot be in P_2 as e_2 is in P_2^i . Consider another edge e_1'' falling under rule R3. $H_1(e_1'')$ cannot contain a as we would get $e_1'' = e$ which cannot follow rule R3. If $H_1(e_1'')$ contains e_2 , we fall back into Case 1.

Case 2 may occur both at u_{i-1} and u_i . All other edges falling under rule R1 are contained in at most one replacement path of an edge following R3. We thus have $x_3 \leq x_1$ in any case. □

Lemma 5 *F has at most $3d_G^2(u, v)$ edges.*

Proof. As $|F^i| \leq 3x_0 + x_1 + 3x_2 + 5x_3$ and $x_3 \leq x_1$,

$$|F^i| \leq 3x_0 + x_1 + 3x_2 + 2x_3 + 3x_3 \leq 3x_0 + 3x_1 + 3x_2 + 3x_3 \leq 3|P^i|$$

As F is the union of all F^i , F has a maximum weight of $3d_G^2(u, v)$. □

All these lemmas (1–5) allow us to conclude with Theorem 2. Indeed, for a pair of original paths in G going from u to v and of length $d_G^2(u, v)$, we are able to build a new subgraph F of H . Thanks to Lemma 2 we show that it contains 2 disjoint paths. Lemma 5 bounds its size. So H contains 2 disjoint paths going from u to v and of size at most $3 \cdot d_G^2(u, v)$. □

Theorem 2 improves the stretch bound of Theorem 1 for the case of unweighted edge-disjoint 2-multipath spanners with $O(n^3/2)$ edges.

We know (from the bipartite complete graph $K_{n/2, n/2}$) that with $O(n^3/2)$ edges a subgraph cannot have an integer multiplicative stretch less than 3. Thanks to Proposition 2 we know this translates to 2-multipath spanners. Therefore Theorem 2 is optimal in the case of 2-paths.

However there exist spanner algorithms (such as the (1, 2)-spanner algorithm from [ACIM99]) which yield lower multiplicative stretch. The next section examines how it is possible to obtain a similar result for 2-path spanners.

3.2.4 A 2-multipath $(2, 8W)$ -spanner

So far, only multiplicative multipath spanners have been considered. In this section, we try to design an algorithm similar to the one used in [ACIM99] to create additive multipath (1, 2)-spanners. This will allow us to create multipath spanners with multiplicative stretch less than the multiplicative stretch of 3 obtained so far.

3.2.4.1 Shortest 2-path spanning trees

To prove the main result of this section we extend the notion of spanning tree.

A p -multipath spanning tree of G is a subgraph T of G with a distinguished vertex u , called the *root* of T , such that, for every vertex v of G , T contains a p -path from u to v . Moreover, T is a p -shortest-path spanning tree if $d_T^p(u, v) = d_G^p(u, v)$ for every vertex v . For $p = 1$, we come back to the standard notions of spanning tree and shortest-path spanning tree. Observe that T may not exist, for instance, if G is not 2-edge-connected.

Lemma 6 *Every n -vertex 2-edge-connected graph with a given vertex u has a 2-shortest-path spanning tree rooted at u with at most $2(n - 1)$ edges constructible in the time of two Dijkstra's shortest path calculations.*

Proof. We use the construction of [ST84]. Their algorithm builds a structure which answers the problem of finding pairs of shortest paths from a single source s to every other vertex. Initially the result in [ST84] is for asymmetric directed graphs but the authors state in section IV that their algorithm can be easily extended to work with any directed graph with multiple edges.

To this effect transform the graph G into G' as follows. Every undirected edge $e = uv$ is replaced by two directed edges: $e_1 = u \leftarrow v$ and $e_2 = u \rightarrow v$.

The algorithm from [ST84] can then be run on G' , using the small modification from section IV.

The algorithm is in essence two Dijkstra's shortest-path tree-like computations. These two calculations build a structure which can answer the 2-distance from the source s to any other vertex. The two paths are retrieved via a specific traversal procedure. To get the real paths in G from those of G' , one needs simply to discard edges used in both directions.

The structure consists of a shortest path tree rooted in s (called T in the article) and on every vertex u another designated edge $u - p(u)$ distinct from the one from T . As the two edges are sufficient to build the two paths, it means that a 2-shortest path tree can be built using at most $2 \cdot |V(G)|$ for any graph G . \square

The bound of $2(n - 1)$ is tight because of the graph $K_{2,n-2}$. More generally, the number of edges in any p -multipath spanning tree must be, in the worst-case, at least $p(n - p)$, for every $p \leq n/2$. Indeed, every p -multipath spanning tree T must be p -edge-connected², and the graph $K_{p,n-p}$ is minimal for the p -edge-connectivity. Therefore, T contains all the edges of $K_{p,n-p}$, and there are $p(n - p)$ edges. Obviously, there are p -multipath spanning tree with less than

²By Proposition 1, there are two edge-disjoint paths between any two vertices of the p -multipath tree, through its root.

$p(n - p)$ edges. Typically a subdivision of $K_{2,p}$ with n vertices has p -multipath spanning tree rooted at a degree- p vertex with a total of $n + p - 2$ edges.

3.2.4.2 A stretch-(2,8W) spanner

Theorem 3 *Every multi-edge weighted graph with n vertices and largest edge-weight W has an edge-disjoint 2-multipath $(2, 8W)$ -spanner with at most $\Phi n^{3/2} + n$ edges, where $\Phi \approx 1.618$ is the golden ratio.*

Proof. Let denote by $B_H^p(u, r) = \{v \in V(H) : d_H^p(u, v) \leq r\}$ the edge-disjoint p -multipath ball of radius r in H centred at u , and denote by $N_H^p(u, r)$ the neighbors of u in H that are in $B_H^p(u, r)$. Note that for $p \geq 2$, some neighbor v of u might not be in $B_H^p(u, r)$ for every $r < \infty$: for instance if u and v are not in the same 2-edge-connected component. We denote by $\text{SPT}_H^p(u)$ a p -shortest-path tree rooted at u spanning the 2-edge-connected component of H containing u , and having at most $2(|E(H)| - 1)$ edges. According to Lemma 6, such p -shortest path tree can be constructed in polynomial time.

Let G be a multi-edge weighted graph with n vertices and largest edge-weight W . We denote by ω its edge-weight function. The 2-multipath spanner H of G is constructed thanks to the following algorithm (see Algorithm 1).

1. For each edge e of G : if there are in G two other edges between the endpoints of e with weight at most $\omega(e)$, then $G := G \setminus \{e\}$
2. $H := (V(G), \emptyset)$ and $W := \max \{\omega(e) : e \in E(G)\}$
3. While there exists $u \in V(G)$ such that $|N_G^2(u, 4W)| \geq (\sqrt{5} - 1)\sqrt{n}$:
 - (a) $H := H \cup \text{SPT}_G^2(u)$
 - (b) $G := G \setminus N_G^2(u, 4W)$
 - (c) $W := \max \{\omega(e) : e \in E(G)\}$
4. $H := H \cup G$

Algorithm 1: An edge-disjoint 2-multipath $(2, 8W)$ -spanner algorithm.

Size. Denote by G_3 and H_3 respectively the graphs G and H obtained after running the while-loop. Let b be the number of while-loops performed by the algorithm, and let $a = \sqrt{5} - 1$. Observe that $a^2 + 2a = 4$. From Lemma 6, the 2-shortest-path tree $\text{SPT}_G^2(u)$ has at most $2(n - 1)$ edges. Hence, the size of H_3 is at most

$$|E(H_3)| < 2b \cdot n.$$

The number of vertices of G_3 is at most $n - ab\sqrt{n}$, since at each loop, at least $a\sqrt{n}$ vertices are removed from G . Let G_3^1 be the graph induced by all the edges (u, v) of G_3 such that $v \in N_{G_3}^2(u, 4W_1)$, where W_1 is the maximum weight of an edge of the graph obtained after running Instruction 1. Let G_3^2 be the graph induced by the edges of $G_3 \setminus G_3^1$. The degree of each vertex u of G_3^1 is $|N_{G_3}^2(u, 4W_1)| - 1$ which is $< \lceil a\sqrt{n} \rceil - 1 \leq a\sqrt{n}$ because of the while-condition. Therefore, the size of G_3^1 is at most

$$|E(G_3^1)| \leq \frac{1}{2} \sum_{u \in V(G_3)} a\sqrt{n} < \frac{1}{2} (n - ab\sqrt{n}) \cdot a\sqrt{n} < \frac{a}{2} \cdot n^{3/2} - \frac{a^2b}{2} \cdot n.$$

Let S_3 be the graph obtained from G_3^2 where each multi-edge is replaced by a single unweighted edge. More formally, vertices u and v are adjacent in S_3 if and only if there is at least one edge between u and v in G_3^2 . From Instruction 1, there are at most two edges between two adjacent vertices, so $|E(G_3^2)| \leq 2|E(S_3)|$.

Let us show that S_3 has no cycle of length ≤ 4 . Consider any edge (u, v) of S_3 . Observe that by the definition of G_3^1 and G_3^2 , $v \notin N_{G_3}^2(u, 4W_3)$ where W_3 is the maximum weight of an edge of G_3 . Assume there is a path cycle of length at most 4 in S_3 going through (u, v) . Then in G_3^2 there is a 2-path from u to v of cost at most $4W_3$. Contradiction: $v \notin N_{G_3}^2(u, 4W_3)$ implies $d_{G_3^2}^2(u, v) > 4W_3$.

It has been proved in [AHL02] that every simple η -vertex μ -edge graph without cycle of length $\leq 2k$, must verify the Moore bound:

$$\eta \geq 1 + \delta \sum_{i=0}^{k-1} (\delta - 1)^i > (\delta - 1)^k$$

where $\delta = 2\mu/\eta$ is the average degree of the graph. This implies that $\mu < \frac{1}{2}(\eta^{1+1/k} + \eta)$.

By definition, S_3 is simple. It follows, for $k = 2$ and $\eta \leq n - ab\sqrt{n}$, that the size of G_3^2 is at most (twice the one of S_3):

$$|E(G_3^2)| \leq (n - ab\sqrt{n})^{3/2} + n - ab\sqrt{n} < (n - ab\sqrt{n})\sqrt{n} + n = n^{3/2} + (1 - ab) \cdot n.$$

Overall, the total number of edges of the final spanner H is bounded by:

$$\begin{aligned} |E(H)| &\leq |E(H_3)| + |E(G_3^1)| + |E(G_3^2)| \\ &< \left(1 + \frac{a}{2}\right) \cdot n^{3/2} + \left(2b - \frac{a^2b}{2} + 1 - ab\right) \cdot n \\ &= \left(1 + \frac{a}{2}\right) \cdot n^{3/2} + n = \frac{1 + \sqrt{5}}{2} \cdot n^{3/2} + n = \Phi n^{3/2} + n \end{aligned}$$

because $2b - a^2b/2 + 1 - ab = b/2 \cdot (4 - a^2 - 2a) + 1 = 1$. (Recall that, by the choice of a , $a^2 + 2a = 4$.)

Stretch. Let G_0 be the input graph G , before applying the algorithm. We first observe that we can restrict our attention to the stretch analysis of G_1 (instead of G_0), the graph obtained after applying Instruction 1.

Indeed, let H be an edge-disjoint 2-multipath spanner for G_1 . Consider two vertices u, v of H , and let A be a minimum-cost 2-path between u and v in H . Since A is composed of two edge-disjoint paths and is of minimum cost in H , A traverses (at most) two edges with same end vertices. Among all pairs of edges from G_0 these two edges have the smallest weight. Therefore these (possibly) two edges exist in both G_0 and G_1 . Hence the 2-multipath stretch of H in G_0 or in G_1 is the same.

From the above observation, it suffices to prove that H is a 2-multipath $(2, 8W_1)$ -spanner of G_1 , where $W_1 \leq W_0$ is the maximum weight of an edge of G_1 .

Let x, y be any two vertices of G_1 , and A be a minimum-cost 2-path between x and y in G_1 . Hence $\omega(A) = d_{G_1}^2(x, y)$. If all the edges of A are in H , then $d_H^2(x, y) = d_{G_1}^2(x, y) = d$, and the stretch is $(1, 0)$. So, assume that $A \not\subseteq H$. Let u be the first vertex selected in the while-loop such that $N_G^2(u, 4W)$ intersects A , so that Instruction 3(b) removes at least one edge of A . Let G, H be the graphs at the time u is selected, but before running Instruction 3(a) and 3(b). Let $v \in N_G^2(u, 4W) \cap A$, and B a minimum-cost 2-path from u to v in G . By definition of $N_G^2(u, 4W)$, $d_G^2(u, v) = \omega(B) \leq 4W$. Let $T = \text{SPT}_G^2(u)$.

An important observation is that u, x, y are in the same 2-edge-connected component of G . This comes from the fact that every 2-path is a 2-edge-connected subgraph³. So, A and B are 2-edge-connected, and $A \cup B$ as well, since A intersects B (in v). Hence T reaches both x and y .

Using the triangle inequality (Proposition 1) in H , we have $d_H^2(x, y) \leq d_H^2(u, x) + d_H^2(u, y)$. By construction of H and T , $d_H^2(u, x) = d_T^2(u, x) = d_G^2(u, x) \leq \omega(A \cup B) \leq \omega(A) + \omega(B)$ since T reaches both x and y . Thus, $d_H^2(u, x) \leq \omega(A) + 4W_1$. Similarly, $d_H^2(u, y) \leq \omega(A) + 4W_1$. Therefore, $d_H^2(x, y) \leq 2d + 8W_1$. The subgraph H is an edge-disjoint 2-multipath $(2, 8W_1)$ -spanner, completing the proof. \square

Theorem 3 shows that it's possible to build spanners with an additive component and a small multiplicative constant, thereby improving the stretch from Theorem 1. Still there is room for improvement. Figure 3.5 does not tell anything about the possibility of building spanners with an even smaller multiplicative constant. It only says that the approach devised in algorithm 1 will not yield better spanners than stated in Theorem 3.

All these improvements results (Theorem 2 and Theorem 3) over Theorem 1

³This observation becomes wrong whenever p -paths with $p > 2$ are considered.

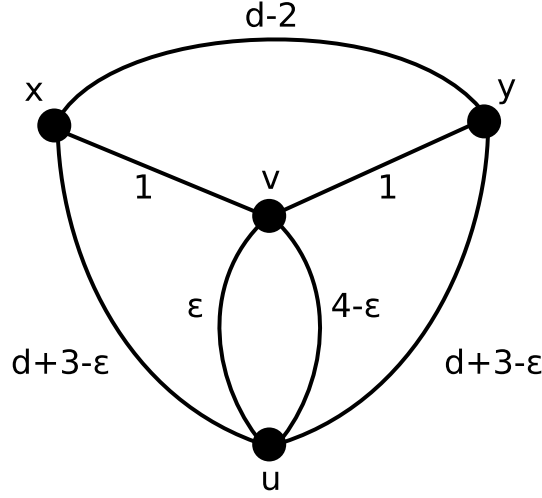


Figure 3.5: A weighted graph G with $d_G^2(x, y) = d$ showing that the stretch analysis in the proof of Theorem 3 is tight. The 2-shortest-path tree rooted at u spans all the edges but (x, y) . We have $d_T^2(u, x) = d_T^2(u, y) = d + 4$, and $d_T^2(x, y) = 2d + 8 - 2\varepsilon$.

leave open the question of having better stretches than those of Theorem 1 for p -multipath spanners with $p > 2$. The next section answers this question.

3.2.5 From edge-fault tolerant spanners to multipath spanners

In this section, we observe that the simple analysis of section 3.2.1 can be improved: we show that f -edge-fault tolerant $(2k - 1)$ -spanners can be used as edge-disjoint p -multipath $(2k - 1)$ -spanners, provided that the number of faults f is much greater than p (polynomial in p, k and logarithmic with respect to the maximal edge weight W). This results in the removal of the factor p in the stretch compared to the one obtained in Theorem 1.

Fault-tolerant spanners were introduced for general graphs in [CLPR10], and were defined as follows:

Definition 5 ([CLPR10]) H is an edge f -fault tolerant α -spanner of G if for every $F \subset E(G)$ such that $|F| \leq f$, the graph H' defined as the graph H minus the edges of F is an α -spanner of the graph G minus the same set of edges F .

The authors of [CLPR10] show that iterative spanners (see Definition 4) are edge-fault-tolerant spanners.

In the following, we show that by adjusting the number of faults (to a number larger than p), one can obtain an edge-disjoint p -multipath $(2k - 1)$ -spanner:

Theorem 4 *Given a weighted graph G with maximal edge weight W and minimal edge weight 1, integers k, p , one can efficiently construct an edge-disjoint p -multipath $(2k - 1)$ -spanner with $O(k \cdot p^2 \cdot \log(W) \cdot n^{1+\frac{1}{k}})$ edges.*

In the following, we use the notion of s -bypass. An s -bypass for an edge $e = uv$ is a path of length at most s going from u to v , avoiding the edge e . A set of s -bypasses \mathcal{B} for an edge e will be called an *edge-disjoint* set of s -bypasses if every path of \mathcal{B} is a bypass and is edge-disjoint from all other paths of \mathcal{B} . Such a set \mathcal{B} will be called a set of *E-bypasses*.

For a path P and vertices $a, v \in P$, we will denote $P' = P[a, b]$ its sub-path P' going from vertex a to vertex b .

To prove Theorem 4 we first concentrate on the unweighted case.

3.2.5.1 Unweighted graphs

The idea of the proof presented below is as follows: for a pair of vertices x, y , consider a p -path of minimum cost between x and y , called $\mathcal{S} = \{Q_1 \cup \dots \cup Q_p\}$ in the rest of the proof. For an edge $e = uv$ belonging to some Q_i of the multipath, we consider all possible single paths of length at most s that bypass this edge. Lemma 7 shows that only a small number of these bypasses will intersect with other edges situated on the other paths composing the original multipath between x and y .

As the statement of Lemma 7 is completely generic, it also applies to any subgraph. If we have a subgraph which guarantees enough bypasses for some edge of the original multipath between x and y , we know by Lemma 7 that there will be one which is disjoint from the edges of other paths of \mathcal{S} .

We next note that an iterative spanner H provides a bypass in every sub-spanner if the edge e does not appear in $E(H)$.

We then divide the iterative spanner into pools, one for each path Q_i . If each pool is sufficiently large (according to Lemma 7), then for each edge missing in the spanner H we will find a bypass in one of the i -th pool's sub-spanner which does not intersect with other edges from $Q_j, j \neq i$, and by construction of the spanner this bypass won't intersect with a bypass from another pool.

Lemma 7 *Consider two vertices x and y and let $\mathcal{S} = \{Q_1, \dots, Q_p\}$ be a p -edge-disjoint multipath of G going from x to y of minimal cost. Consider an edge $e = uv \in Q_i$ for some $1 \leq i \leq p$. Then in G there are at most $2sp + 2p$ edge-disjoint s -bypasses of the edge e that intersect with other edges from other paths $Q_j, j \neq i$.*

Proof. The proof is done by contradiction.

Suppose that there are more than $2sp + 2p$ edge-disjoint bypasses of length at most s between u and v that intersect with edges from paths of \mathcal{S} other than

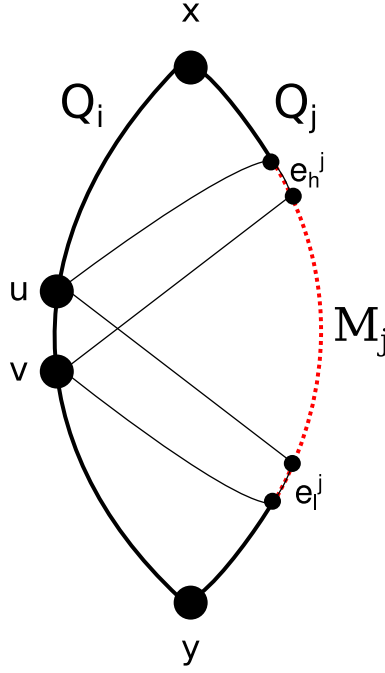


Figure 3.6: Definition of M_j , e_h^j and e_l^j . M is the reunion of all M_j

Q_i . Let \mathcal{B} be the set of all these bypasses that intersect with edges from the other paths from \mathcal{S} . Let J be the set of indices $1 \leq j \leq p, j \neq i$ such that there is a bypass from \mathcal{B} which intersects with an edge of Q_j : $E(Q_j) \cap E(\mathcal{B}) \neq \emptyset$. On each path Q_j , call $e_h^j = (u_h^j, v_h^j)$ and $e_l^j = (u_l^j, v_l^j)$ the uppermost (closest from x) edge and lowermost (closest from y) edge belonging to both Q_j and a path from \mathcal{B} . For each path Q_j , let $M_j = Q_j[u_h^j, v_l^j]$ be its sub-path from u_h^j to v_l^j , and let $M = \bigcup_{j \in J} M_j$. This naming convention is shown on Fig. 3.6.

We show that it is possible to replace the set of edges of M with a set of edges E' such that the resulting graph contains p edge disjoint paths from x to y and its cost is less than $\omega(\mathcal{S})$ and thus derive a contradiction to the optimality of \mathcal{S} .

We now explain how to gradually build the set of edges E' . Roughly speaking, the set of edges E' is the union of some prefixes and suffixes of the paths M_j together with some edge-disjoint bypasses from \mathcal{B} .

Let p_j be the prefix of M_j , and s_j the suffix of M_j . Initially these are set to the uppermost edge ($p_j = e_h^j$) and lowermost edge ($s_j = e_l^j$). The construction process will gradually add edges to these prefixes and suffixes.

Furthermore, let $\text{tip}(p_j)$ (resp., $\text{tip}(s_j)$) be the edge in $E(p_j) \cap E(\mathcal{B})$ (resp., $E(s_j) \cap E(\mathcal{B})$) closest to y on Q_j (resp., to x on Q_j).

Let $\mathcal{B}' \subset \mathcal{B}$ be the subset of edge-disjoint bypasses in which the tips appear. We examine next this subset as well as the set of tips. Let X be the set of prefixes

and suffixes, namely, $X = \bigcup_{j \in J} \{p_j, s_j\}$. The set \mathcal{B}' is the set of edge disjoint bypasses containing one of the edges $\{\text{tip}(p_j), \text{tip}(s_j) \mid j \in J\}$. For an edge $e \in E(\mathcal{B}')$, let $B(e)$ be the bypass $B \in \mathcal{B}'$ such that $e \in E(B)$. Note that there is exactly one such bypass since the bypasses are disjoint. We say that a path $P \in X$ is *clean* if the sub-path $B(\text{tip}(e))[\text{tip}(P), u]$ does not contain other edges from $E(X)$ ⁴. In other words, the path P is clean when the part of the bypass going from $\text{tip}(P)$ to u does not encounter a tip from another path. For a bypass $B \in \mathcal{B}'$, let $P_{\text{clean}}(B)$ be the path $P \in X$ such that $\text{tip}(P) \in E(B)$ and P is clean; note that there is exactly one such path.

We say that a prefix $p_j \in X$ (resp., suffix s_j) is *complete* if $p_j \circ s_j = M_j$. We apply the following process until all paths in X are clean. Choose an unclean incomplete path and add edges to it until it becomes clean. By adding an edge to a prefix p_j (resp., suffix s_j) we mean adding the edge on M_j adjacent to $\text{tip}(p_j)$ (resp., $\text{tip}(s_j)$) closest to y (resp., to x).

Note that it could happen that during this process some clean path becomes unclean. Note also that bypasses are only added to \mathcal{B}' (but never removed). Namely, $\mathcal{B}'(t_1) \subseteq \mathcal{B}'(t_2)$ for $t_1 \leq t_2$, where $\mathcal{B}'(t)$ is the set \mathcal{B}' in the t 'th step of this process. To see this, note that the process does not add edges to $P_{\text{clean}}(B)$ for any $B \in \mathcal{B}'$. Thus in any stage of this process, B contains $\text{tip}(P)$ for $P = P_{\text{clean}}(B)$. Hence, by definition, $B \in \mathcal{B}'$. Notice that the path $P_{\text{clean}}(B)$ itself may change (since the process might add an edge to another path and this edge could belong to the path from $\text{tip}(P)$ to u).

We claim that \mathcal{B}' contains at most $2p$ bypasses. This follows directly from the fact that each bypass $B \in \mathcal{B}'$ contains a different path $P_{\text{clean}}(B)$ and that there are $2p$ paths in X .

We now show that it is possible to substitute the paths in M with “cheaper” paths and thus derive a contradiction to the optimality of \mathcal{S} . For every incomplete prefix $p_j \in X$, let p'_j be the clean sub-path $B(p_j)[\text{tip}(p_j), u]$. Similarly, let s'_j be the clean sub-path $B(s_j)[u, \text{tip}(s_j)]$. For every index $j \in J$, if p_j is complete then set $Q'_j = Q_j$, otherwise set $Q'_j = Q_j[x, u_h^j] \circ p_j \circ p'_j \circ s'_j \circ s_j \circ Q_j[v_l^j, y]$.

Let $D' = \bigcup_{j \in J} E(p'_j \circ s'_j)$ and $D = \bigcup_{j \neq i} E(M_j) \setminus (E(p_j) \cup E(s_j))$. Let $\mathcal{S}' = \{Q'_1, \dots, Q'_p\}$. Note that $|E(\mathcal{S}')| = |E(\mathcal{S})| + |D'| - |D|$. It is not hard to verify that the paths Q'_j are disjoint and each of them leads from x to y . Moreover, M intersects with at least $2sp + 2p + 1$ edge-disjoint bypasses, and the set $E(p_j) \cup E(s_j)$ intersects with at most $2p$ edge-disjoint bypasses, thus $|D| > 2sp + 1$. In addition, $|D'| \leq 2sp$. We get that $|D| > |D'|$ and thus $\omega(\mathcal{S}') < \omega(\mathcal{S})$, contradiction. \square

We now show that edge fault tolerant spanners constructed with iterative

⁴Note that $\text{tip}(P)$ is always defined for $P \in X$.

spanners (see definition 4) (proof of fault-tolerance in [CLPR10]) are also edge multipath spanners with the same stretch, by fixing the right number of faults.

We summarize in the following lemma the properties a fault tolerant algorithm needs to fulfill in order to be a multipath spanner by our proof.

Lemma 8 ([CLPR10]) *For any weighted graph $G = (V, E)$, integers $k, q \geq 1$, one can construct in polynomial time a collection of edge disjoint subgraphs $\{H_1, \dots, H_q\}$ with the following properties. Let H be the union of the subgraphs $\{H_1, \dots, H_q\}$.*

- (1) *The number of edges in H is at most $O(q \cdot n^{1+\frac{1}{k}})$.*
- (2) *For every edge $e = (u, v) \in E$, either $e \in E(H)$ or each H_i contains a path from u to v of length at most $s = 2k - 1$.*

Lemma 9 *Given an unweighted graph G , integers $p, k \geq 1$, one can efficiently construct a p -multipath $(2k - 1)$ -spanner with $O(kp^2 \cdot n^{1+\frac{1}{k}})$ edges.*

Proof. Let $s = 2k - 1$. Construct the collection of subgraphs $\{H_1, \dots, H_{2sp^2+2p^2+p}\}$ of Lemma 8 with parameters s and $q = 2sp^2 + 2p^2 + p$. Let H be the union of all subgraphs $\{H_1, \dots, H_{2sp^2+2p^2+p}\}$. Consider two vertices x and y and let $\mathcal{S} = \{Q_1, \dots, Q_p\}$ be the set of p edge-disjoint paths from x to y in G of minimal cost (i.e. $\omega(\mathcal{S}) = d^p(x, y)$). We now show how to find a set of edge-disjoint paths $\mathcal{S}' = \{Q'_1, \dots, Q'_p\}$ from x to y such that $E(\mathcal{S}') \subseteq E(H)$ and $\omega(Q'_i) \leq s \cdot \omega(Q_i)$. Let $T_i = \{H_j \mid (2sp + 2p + 1) \cdot (i - 1) + 1 \leq j \leq (2sp + 2p + 1) \cdot i\}$ for $1 \leq i \leq p$. Note that $E(T_i)$ contains $2sp + 2p + 1$ edge-disjoint paths from u to v for every edge $(u, v) \notin E(H)$. Moreover, the sets $E(T_i)$ are disjoint for $1 \leq i \leq p$.

The path Q'_i is constructed as follows. For every edge $e \in E(Q_i) \cap E(H)$ add e to Q'_i . For every edge $e = (u, v) \in E(Q_i) \setminus E(H)$, consider the set \mathcal{B}_i with the maximum number of E -bypasses from u to v in T_i . By Lemma 7, there are at most $2sp + 2p$ E -bypasses in \mathcal{B}_i that intersect with $E(\mathcal{S}) \setminus E(Q_i)$. Since \mathcal{B}_i contains at least $2sp + 2p + 1$ E -bypasses, at least one E -bypass $B(e) \in \mathcal{B}_i$ does not intersect with $E(\mathcal{S}) \setminus E(Q_i)$. Add $B(e)$ to Q'_i instead of e .

We claim that

- (1) the paths Q'_i for $1 \leq i \leq p$ are edge-disjoint, and
- (2) $\omega(\mathcal{S}') \leq s \cdot \omega(\mathcal{S})$.

To see claim (1), consider an edge $e = (u, v)$ such that $e \in E(Q'_i)$ for some $1 \leq i \leq p$. We consider two cases. The first case is when $e \in E(Q_i)$. Note that e does not appear in any $E(Q_j)$ for $i \neq j$, since the paths in \mathcal{S} are disjoint.

Moreover, e does not appear in any $B(e')$ for $e' \in E(Q_j)$ for some $j \neq i$. To see this, recall that $B(e')$ does not intersect with $E(\mathcal{S}) \setminus E(Q_j)$. The second case is when $e \in B(\tilde{e})$ for some $\tilde{e} \in E(Q_i)$. The E-bypass $B(\tilde{e})$ does not intersect with $E(\mathcal{S}) \setminus E(Q_i)$. Moreover, the E-bypass $B(\tilde{e})$ does not intersect with any $B(e')$ for $e' \in E(Q_j)$ for some $j \neq i$. To see this, recall that $E(B(\tilde{e})) \subseteq E(T_i)$, $E(B(e')) \subseteq E(T_j)$, and $E(T_i) \cap E(T_j) = \emptyset$. It follows that the paths Q'_i are edge-disjoint for $1 \leq i \leq p$.

To see claim (2), note that for every edge $e \in E(Q_i)$, either e itself or an alternative path of length s is added to $E(Q'_i)$. We get that $\omega(Q'_i) \leq s \cdot \omega(Q_i)$. Claim (2) follows. \square

3.2.5.2 Weighted Graphs

We now present the modifications to Lemma 9 needed for weighted graphs. Assume the minimal edge weight is 1 and let W be the maximal edge weight.

Proof of Theorem 4. We now describe the algorithm for constructing an edge-disjoint p -multipath s -spanner. Initially, set $H = (V, \emptyset)$. Consider the graphs $G_i = (V, E_i)$ such that $E_i = \{e \in E \mid 2^{i-1} < \omega(e) \leq 2^i\}$ for every $0 \leq i \leq \lceil \log_2 W \rceil$. Construct the collection of subgraphs $F_i = \{H_1, \dots, H_{4sp^2+2p^2+p}\}$ of Lemma 8 for parameters s and $q = 4sp^2 + 2p^2 + p$ on the graph G_i . Add $E(F_i)$ to H .

We claim that H is an edge-disjoint p -multipath s -spanner. The analysis is very similar to the unweighted case. We now outline the slight changes.

Here we call a set of paths \mathcal{B} a set of *E-bypasses* of two nodes u and v that are connected by an edge if the paths in \mathcal{B} are edge-disjoint paths between u and v of length at most $s \cdot \omega(u, v)$ each.

Consider two vertices x and y and let $\mathcal{S} = \{Q_1, \dots, Q_p\}$ be a set of p edge-disjoint paths from x to y in G of minimal cost (i.e., $\omega(\mathcal{S}) = d_G^p(x, y)$). Consider an edge $e = (u, v) \in Q_i$ for some $1 < i \leq p$. Let i be the index such that $2^{i-1} \leq \omega(e) \leq 2^i$. In Lemma 7 we prove that the graph $(V, E(F_i))$ contains at most $4sp + 4p$ E-bypasses \mathcal{B} from u to v . Note that since the weight of the edges in the E-bypasses \mathcal{B} could be half the weight $\omega(e)$, we double the factor of sp^2 in the number of E-bypasses. The rest of the proof of Lemma 7 is similar to the unweighted case.

The proof of Theorem 9 is also similar to the unweighted case, where for each edge $e = (u, v) \in Q_j$ for some $1 \leq j \leq p$ such that $e \notin E(H)$, we pick an E-bypass from F_i that does not intersect $E(\mathcal{S}) \setminus E(Q_j)$, for i such that $2^{i-1} < \omega(e) \leq 2^i$.

We thus conclude with Theorem 4. \square

Theorem 4 is the final improvement over Theorem 1. This is the only general improvement of this chapter. The other theorems deal with particular cases.

One such direction is the search of lower bounds for multipath spanners. To this effect, we first remark in section 3.2.2 that lower bounds from multipath spanners can be transposed as-it-is to multipath spanners (provided multiple edges are allowed). Then with theorems 2 and 3 we explore the lower bounds of 2-multipath spanners with $O(n^{3/2})$ edges. Theorem 2 answers the question for unweighted graphs and multiplicative stretch while Theorem 3 tries to go below the multiplicative bound in the stretch by paying an additional additive factor to the stretch.

Theorem 4 answers a different question: provided we allow some stretch, what can we say about the number of edges? We show in effect that while it is possible to build edge-disjoint p -multipath spanners of stretch $(2k - 1)$ for every k , we can do so only while paying an extra factor in the number of edges (of the form $k \cdot p^2 \log(W)$) compared to one-path spanners of the same stretch. The theorem's proof has an additional interest in that it relates multipath spanners to fault-tolerant spanners described in [CLPR10].

Chapter 4

Vertex-disjoint multipath spanners

In the previous chapter, we studied edge-disjoint multipath spanners. Despite having edge-disjoint multipaths helps multipath routing tasks, they remain sensitive to node failures or when nodes have bounded total bandwidth. This calls for examination of *vertex-disjoint* multipath spanners. To this effect, we introduce in section 4.1.1 the notion of vertex-disjoint p -multipath graph metric, and try to devise p -vertex-disjoint multipath spanners algorithms in the following sections. The major difference between edge-disjoint multipath spanners and vertex-disjoint multipath spanners is that the so-called vertex-disjoint multipath graph metric doesn't satisfy the triangle inequality (as observed in Remark 1). As a consequence, the adaptation of standard spanner algorithms is much more difficult.

4.1 Preliminaries

We still consider here a (possibly weighted) undirected graph $G = (V, E)$ with weight function ω and maximum weight W . For any subgraph $H = (V', E')$, $V' \subset V$, $E' \subset E$ of G we also define $V(H) = V'$ as the set of vertices of H and $E(H) = E'$ as the set of edges of H .

4.1.1 Multipath graph metric

To build vertex-disjoint p -multipath spanners, we first need to define the graph metric which they use. This graph metric uses vertex-disjoint p -multipaths. What present next the definition of vertex disjoint p -multipaths.

Definition 6 A vertex-disjoint p -path (or multipath) $\mathcal{Q} = \{Q_1, \dots, Q_i, \dots, Q_p\}$ between two vertices a and b is a subgraph composed of p pairwise internally (that is except a and b) vertex disjoint paths: for any $i \neq j$, $V(Q_i) \cap V(Q_j) = \{a, b\}$.

Note that in the case of 2-paths, \mathcal{Q} is simply an elementary cycle between, a and b .

In the rest of this chapter, p -paths are considered vertex disjoint unless noted.

Definition 7 The p -multipath graph metric denoted by ${}^v d_H^p(u, v)$ in a subgraph H is defined as the minimum cost of a p -multipath between u and v , using only edges of H . Furthermore, we set ${}^v d_H^p(u, v) = \infty$ if there are strictly less than p vertex disjoint paths between u and v .

As in Part 3, when the graph is omitted it is supposed to be G .

Remark 1 For $p \geq 2$, ${}^v d^p$ is not a distance.

Proof. Indeed, ${}^v d^p$ does not satisfy the triangle inequality. For instance, take $p = 2$ and consider three vertices u, v, w such that ${}^v d^p(u, v)$ and ${}^v d^p(v, w)$ are finite.

If v is an articulation point, there won't be an elementary cycle between u and w , and so ${}^v d^p(u, w) = \infty$. See fig. 4.1 for an example. \square

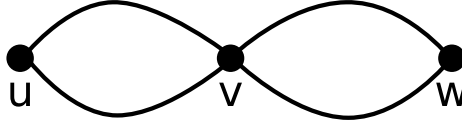


Figure 4.1: ${}^v d^2(u, v)$ and ${}^v d^2(v, w)$ are finite whereas ${}^v d^2(u, w) = \infty$.

All the difficulties to adapt classical spanner algorithms stem from this absence of triangle inequality. Indeed, the analysis of such algorithms often revolve around decomposing a path P_G into smaller subpaths P_G^1, \dots, P_G^k , then by bounding each replacement path P_H^i in the spanner H . If the metric is not endowed with the triangle inequality this approach becomes useless.

4.1.2 Vertex-disjoint multipath spanners

Despite that the graph metric ${}^v d^p$ doesn't have the triangle inequality, the notion of spanner is still relevant. We therefore define a *vertex disjoint p -multipath spanner* as follows:

Definition 8 $G = (V, E)$. $H = (V, E' \subset E)$ is a vertex disjoint p -multipath (α, β) -spanner of G if ${}^v d_H^p(u, v) \leq \alpha \cdot {}^v d_G^p(u, v) + \beta$ for all $u, v \in V$.

When $\beta = 0$, H is simply called an α -spanner.

In the rest of this chapter, we will omit the wording “vertex disjoint” from p -multipath vertex disjoint spanner for simplicity. As in the edge-disjoint case, this definition meets the traditional definition of spanners for $p = 1$.

4.2 Results

In the rest of this chapter, we prove the following results:

- Remark 2: we note that fault tolerant spanners can have an arbitrary large stretch for the $\overset{v}{d}^p$ graph metric.
- Theorem 5: fault-tolerant spanners with an additionnal property are also vertex-disjoint multipath spanners with a bounded stretch.
- Theorem 8 : p -multipath spanners can be built from fault-tolerant spanners provided the number of tolerated faults is polynomial in p . In the case of unweighted graphs, Theorem 7 gives a better stretch.
- Theorem 10: any weighted graph admits a 2-path $(2, O(W))$ -spanner with $O(n^{3/2})$ edges, where W is the maximum edge-weight. The construction relies on an ad-hoc algorithm that takes at most $O(n^4)$ steps.

Remark 2, Theorem 5 and Theorem 10 were published in coöperation with Cyril Gavoille and Laurent Viennot in [GGV11].

Theorem 7 was published in coöperation with Shiri Chechik and David Peleg in [CGP12].

4.2.1 A first remark concerning fault-tolerant spanners

The first question which arises when describing multipath spanners is whether the notion of multipath spanner is related to the notion of fault-tolerant spanner, as defined for general graphs in [CLPR10]:

Definition 9 [CLPR10] *A subgraph H of a graph G is a f -vertex-fault-tolerant s -spanner when for any subset of vertices $F \subset V(G)$, $|F| \leq f$, $\forall u, v \in d_{H \setminus F}(u, v) \leq s \cdot d_{G \setminus F}(u, v)$.*

At first glance, f -fault tolerant spanners seem related to $(f + 1)$ -multipath spanners. Indeed by the Menger theorem two vertices linked by $f + 1$ paths in the graph will also be linked by $f + 1$ paths in the fault-tolerant spanner of the graph. However we have the following remark:

Remark 2 *In the context of weighted graphs, there are 1-fault tolerant s -spanners with arbitrary large stretch for the 2-multipath distance.*

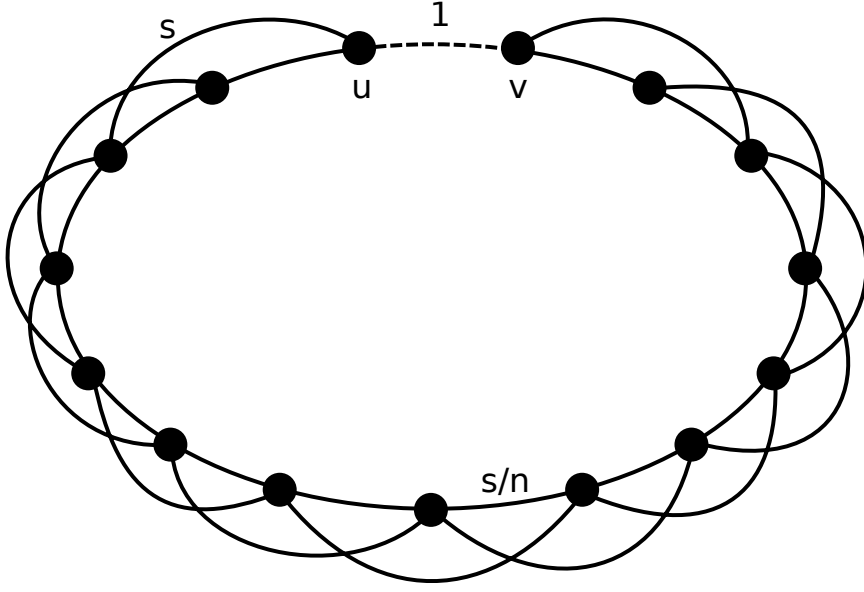


Figure 4.2: H is a 1-fault-tolerant $2s$ -spanner, but of stretch at least n when considered as a 2-multipath spanner for the d^2 graph metric.

Indeed, take figure 4.2. It presents a weighted graph G composed of a cycle of $n + 1$ vertices plus $n - 1$ extra edges. H will be $G \setminus \{uv\}$. The edge uv has weight 1, non cycle edges have weight s and cycle edges have weight s/n so that $d_H(u, v) = s$. Removing any vertex $z \notin \{u, v\}$ implies $d_{G \setminus \{z\}}(u, v) = 1$ and $d_{H \setminus \{z\}}(u, v) = 2s(1 - 1/n)$. For any other pair of vertices, x, y , we have $d_{H \setminus \{z\}}(x, y)/d_{G \setminus \{z\}} < 2s$. Thus H is a 1-fault tolerant $2 \cdot s$ -spanner. However $d_H^2(u, v)/d_G^2(u, v) \geq sn/s$ which implies that H is a 2-multipath spanner with stretch at least n .

Note that this result holds only for weighted graphs.

4.2.2 From fault-tolerant spanners to multipath spanners

Despite the negative previous result, we can still use fault-tolerant spanners to get results on multipath spanners, provided they have additional properties.

In fact, the problem with Remark 2 is that the fault-tolerant spanner does not guarantee that the number of edges composing a path replacing a missing edge is bounded. Provided we have this condition, some results are possible.

In this section, we aim to prove the following result:

Theorem 5 *Let G be a weighted graph with n vertices, and p, k be integral parameters ≥ 1 . Then, G has a vertex-disjoint p -multipath $kp \cdot O(1 + p/k)^{2k-1}$ -spanner of*

size $O(kp^{2-1/k}n^{1+1/k} \log^{2-1/k} n)$ that can be constructed with high probability by a randomized distributed algorithm in $O(k)$ rounds.

Theorem 5 is proved by combining several constructions presented now.

4.2.2.1 Spanners with few hops

The problem with figure 4.2 is that the replacement path for the edge uv has too many hops (or too many edges with a low weight). To avoid this problem and towards proof of Theorem 5, we introduce the concept of *bounded hop spanner*.

Definition 10 An s -spanner H of a weighted graph G is b -hop if for every edge uv of G , there is a path in H between u and v composed of at most b edges and of cost at most $s \cdot \omega(uv)$ (where $\omega(uv)$ denotes the cost of edge uv). An s -hop spanner is simply an s -hop s -spanner.

If G is unweighted (or the edge-cost weights are uniform), the concepts of s -hop spanner and s -spanner coincide.

Note that the spanners produced by the greedy¹ algorithm [ADD⁺93] are not necessarily s -hop.

However, we have:

Proposition 3 For each integer $k \geq 1$, every weighted graph with n vertices has a $(2k - 1)$ -hop spanner with less than $n^{1+1/k}$ edges.

Proof. Consider a weighted graph G with edge-cost function ω . We construct the desired spanner H of G thanks to the following algorithm which can be seen as the dual of the classical greedy algorithm, till a variant of Kruskal's algorithm:

- (1) Initialize H with $V(H) := V(G)$ and $E(H) := \emptyset$;
- (2) Visit all the edges of G in non-decreasing order of their weights, and add the edge uv to H only if every path between u and v in H has more than $2k - 1$ edges.

Consider an edge uv of G . If uv is not in H then there must exist a path P in H from u to v such that P has at most $2k - 1$ edges. We have $d_H(u, v) \leq \omega(P)$. Let e be an edge of P with maximum weight. We can bound $\omega(P) \leq (2k - 1) \cdot \omega(e)$. Since e has been considered before the edge uv (because e participates in the path P), $\omega(e) \leq \omega(uv)$. It follows that $\omega(P) \leq (2k - 1) \cdot \omega(uv)$, and thus

¹For each edge uv in non-decreasing order of their weights, add it to the spanner if $d_H(u, v) > s \cdot d_G(u, v)$.

$d_H(u, v) \leq (2k - 1) \cdot \omega(uv)$. Obviously, if uv belongs to H , then $d_H(u, v) = \omega(uv) \leq (2k - 1) \cdot \omega(uv)$ as well. Therefore, H is a $(2k - 1)$ -hop spanner.

The fact that H is sparse comes from the fact that there is no cycle of length $\leq 2k$ in H : whenever an edge is added to H , any path linking its endpoints has more than $2k - 1$ edges, i.e., at least $2k$.

We observe that H is simple even if G is not. It has been proved in [AHL02] that every simple n -vertex m -edge graph where every cycle is of length at least $2k + 1$ (i.e., of girth at least $2k + 1$), must verify the Moore bound:

$$n \geq 1 + d \sum_{i=0}^{k-1} (d-1)^i > (d-1)^k$$

where $d = 2m/n$ is the average degree of the graph. This implies that $m < \frac{1}{2}(n^{1+1/k} + n) < n^{1+1/k}$.

Therefore, H is a $(2k - 1)$ -hop spanner with at most $n^{1+1/k}$ edges. \square

4.2.2.2 Distributed bounded hop spanners

We now show how to build bounded hop spanners in a distributed manner.

There are distributed constructions that provide s -hop spanners, at the cost of a small (poly-logarithmic in n) increase of the size of the spanner compared to Proposition 3.

If we restrict our attention to deterministic algorithms, [DGPV08] provides for unweighted graphs a $(2k - 1)$ -hop spanner of size $O(kn^{1+1/k})$. It runs in $3k - 2$ rounds without any prior knowledge on the graph, and optimally in k rounds if n is available at each vertex.

Proposition 4 *There is a distributed randomized algorithm that, for every weighted graph G with n vertices, computes with high probability a $(2k - 1)$ -hop spanner of $O(kn^{1+1/k} \log^{1-1/k} n)$ edges in $O(k)$ rounds.*

Proof. The algorithm is a distributed version of the spanner algorithm used in [CLPR10], which is based on the sampling technique of [TZ05]. We make the observation that this algorithm can run in $O(k)$ rounds. Let us briefly recall the construction of [CLPR10, p. 3415].

To each vertex x of G is associated a tree rooted at x spanning the *cluster* of x , a particular subset of vertices denoted by $C(x)$. The construction of $C(x)$ is a refinement over the one given in [TZ05]. The main difference is that the clusters' depth is no more than k edges. The spanner is composed of the union of all such cluster spanning trees. The total number of edges is $O(kn^{1+1/k} \log^{1-1/k} n)$.

It is proved in [CLPR10] that for every edge uv of G , there is a cluster $C(x)$ containing u and v . The path of the tree from x to one of the end-point has at most $k - 1$ edges and cost $\leq (k - 1) \cdot \omega(uv)$, and the path from x to the other end-point has at most k edges and cost $\leq k \cdot \omega(uv)$. This is therefore a $(2k - 1)$ -hop spanner.

The random sampling of [TZ05] can be done without any round of communications, each vertex randomly selects a level independently of the other vertices. Once the sampling is performed, the clusters and the trees can be constructed in $O(k)$ rounds as their depth is at most k . \square

4.2.2.3 Fault tolerant spanners

In this section we show how to use known results concerning fault tolerant spanners (mainly [DK11]) in order to build bounded hop fault tolerant spanners. The algorithm of [DK11] for constructing fault tolerant spanners is randomized and generic. It takes as an input a weighted graph G with n vertices, a parameter $r \geq 0$, and any algorithm **A** computing an s -spanner with $m(\nu)$ edges for any ν -vertex subgraph of G . With high probability, it constructs for G an r -fault tolerant s -spanner of size $O(r^3 \cdot m(2n/r) \cdot \log n)$. It works as follows:

Set $H := \emptyset$, and repeat independently $O(r^3 \log n)$ times:

- (1) Compute a set S of vertices obtained by selecting each vertex with probability $1 - 1/(r + 1)$;
- (2) $H := H \cup \mathbf{A}(G \setminus S)$.

Then, they show that for every fault set $F \subset V(G)$ of size at most r , and every edge uv , there exists with high probability a set S computed as in Step (1) for which $u, v \notin S$ and $F \subseteq S$. As a consequence, routine $\mathbf{A}(G \setminus S)$ provides a path between u and v in $G \setminus S$ (and thus also in $G \setminus F$) of cost $\leq s \cdot \omega(uv)$. If uv lies on a shortest path of $G \setminus F$, then this cost is $\leq s \cdot d_{G \setminus F}(u, v)$. From their construction, we have:

Proposition 5 *If **A** is a distributed algorithm constructing an s -hop spanner in t rounds, then algorithm [DK11] provides a randomized distributed algorithm that in t rounds constructs with high probability an s -hop r -fault tolerant spanner of size $O(r^3 \cdot m(2n/r) \cdot \log n)$.*

Proof. The resulting spanner H is s -hop since either the edge uv of G is also in H , or a path between u and v approximating $\omega(uv)$ exists in some s -hop spanner given by algorithm **A**. This path has no more than s edges and cost $\leq s \cdot \omega(uv)$.

Observe that the algorithm [DK11] consists of running in parallel $q = O(r^3 \log n)$ times independent runs of algorithm **A** on different subgraphs of G , each one using t rounds. Round i of all these q runs can be done into a single round of communication, so that the total number of rounds is bounded by t , not by q .

More precisely, each vertex first selects a q -bit vector, each bit set with probability $1 - 1/(r + 1)$, its j th bit indicating whether it participates to the j th run of **A**. Then, q instances of algorithm **A** are run in parallel simultaneously by all the vertices, and whenever the algorithms perform their i th communication round, a single message concatenating the q messages is sent. Upon reception, a vertex expands the q messages and run the j th instance of algorithm **A** only if the j th bit of its vector is set.

The number of rounds is no more than t . □

4.2.2.4 From fault tolerant to multipath spanner

We now present the main result of this section:

Theorem 6 *Let H be a s -hop $(p - 1)$ -fault tolerant spanner of a weighted graph G . Then, H is also a p -multipath $\varphi(s, p)$ -spanner of G where $\varphi(s, p) = sp \cdot O(1 + p/s)^s$ and $\varphi(3, p) = 9p$.*

The idea of the proof is as follows: for an edge e of the graph not in the spanner, consider the replacement path in the spanner for this edge. The fault-tolerant property states that for any set of faults of size less than $p - 1$ there is still a path of length at most $s \cdot \omega(e)$. Grow recursively the set of paths by doing every possible fault set on the paths. Lemma 10 bounds the size of the set of paths obtained by this manner. Then Theorem 6 extends the result to any pair of vertices instead of one linked by an edge.

To prove Theorem 6, we need the following intermediate result, assuming that H and G satisfy the statement of Theorem 6.

Lemma 10 *Let uv be an edge of G of weight $\omega(uv)$ that is not in H . Then, H contains a p -multipath connecting u to v of cost at most $\varphi(s, p) \cdot \omega(uv)$ where $\varphi(s, p) = sp \cdot O(1 + p/s)^s$ and $\varphi(3, p) = 9p$.*

Proof. From Menger's Theorem, the number of pairwise vertex-disjoint paths between two non-adjacent vertices x and y equals the minimum number of vertices whose removal disconnects x and y .

By definition of H , $H \setminus F$ contains a path P_F of at most s edges between u and v for each set F of at most $p - 1$ vertices (excluding u and v). This is because

u and v are always connected in $G \setminus F$, precisely by a single edge path of cost $\omega(uv)$. Consider P_H the subgraph of H composed of the union of all such P_F paths (so from u to v in $H \setminus F$ – see Fig. 4.3 for an example with $p = 2$ and $s = 5$).

Vertices u and v are non-adjacent in P_H . Thus by Menger's Theorem, P_H has to contain a p -multipath between u and v . Ideally, we would like to show that this multipath has low cost. Unfortunately, Menger's Theorem cannot help us in this task.

Let $\kappa_s(u, v)$ be the minimum number of vertices in P_H whose deletion destroys all paths of at most s edges between u and v , and let $\mu_s(u, v)$ denote the maximum number of internally vertex-disjoint paths of at most s edges between u and v . Obviously, $\kappa_s(u, v) \geq \mu_s(u, v)$, and equality holds by Menger's Theorem if $s = n - 1$. Equality does not hold in general as presented in Fig. 4.3. However, equality holds if s is the minimum number of edges of a path between u and v , and for $s = 2, 3, 4$ (cf. [LNLP78]).

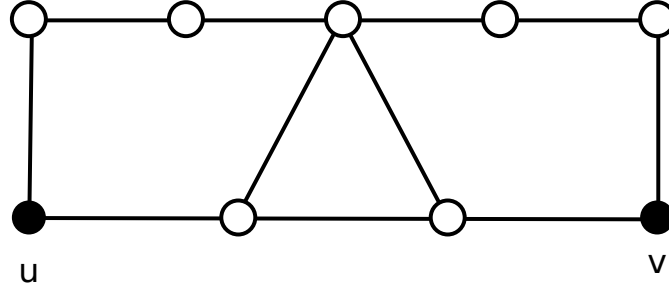


Figure 4.3: A subgraph P_H constructed by adding paths between u and v with at most $s = 5$ edges and with $p = 2$. Removing any vertex leaves a path of at most 5 edges, so $\kappa_5(u, v) > 1$. However, there aren't two vertex-disjoint paths from u to v of at most 5 edges, so $\kappa_5(u, v) > \mu_5(u, v)$. Observe that $\mu_6(u, v) = \kappa_5(u, v) = 2$.

Since not every path of at most s edges between u and v is destroyed after removing $p - 1$ vertices in P_H , we have that $\kappa_s(u, v) \geq p$. Let us bound the total number of edges in a p -multipath Q of minimum size between u and v in P_H . Let r be the least number such that $\mu_r(u, v) \geq p$ subject to $\kappa_s(u, v) \geq p$. The total number of edges in Q is therefore no more than pr .

By construction of P_H , each edge of P_H comes from a path in $H \setminus F$ of cost $\omega(P_F) \leq s \cdot d_{G \setminus F}(u, v) \leq s \cdot \omega(uv)$. In particular, each edge of Q has weight at most $s \cdot \omega(uv)$. Therefore, the cost of Q is $\omega(Q) \leq prs \cdot \omega(uv)$.

It has been proved in [PT93] that r can be upper bounded by a function $r(s, p) < \binom{p+s-2}{s-2} + \binom{p+s-3}{s-2} = O(1 + p/s)^s$ for integers s, p , and $r(3, p) = 3$ since as seen earlier $\kappa_3(u, v) = \mu_3(u, v)$. It follows that H contains a p -multipath Q between u and v of cost $\omega(Q) \leq sp \cdot O(1 + p/s)^s \cdot \omega(uv)$ as claimed. \square

Proof of Theorem 6. Let x, y be any two vertices of a graph G with edge-cost function ω . We want to show $\omega_H^p(x, y) \leq \varphi(s, p) \cdot \omega_G^p(x, y)$. If $\omega_G^p(x, y) = \infty$, then we are done. So, assume that $\omega_G^p(x, y) = \omega(P_G)$ for some minimum cost p -multipath P_G between x and y in G . Note that $\omega(P_G) = \sum_{uv \in E(P_G)} \omega(uv)$.

We construct a subgraph P_H between x and y in H by adding: (1) all the edges of P_G that are in H ; and (2) for each edge uv of P_G that is not in H , the p -multipath Q_{uv} connecting u and v in H as defined by Lemma 10.

The cost of P_H is therefore:

$$\begin{aligned} \omega(P_H) &= \sum_{uv \in E(P_H)} \omega(uv) \\ &= \left(\sum_{uv \in E(P_G) \cap E(H)} \omega(uv) \right) + \left(\sum_{uv \in E(P_G) \setminus E(H)} \omega(Q_{uv}) \right) \end{aligned}$$

By Lemma 10, $\omega(Q_{uv}) \leq \varphi(s, p) \cdot \omega(uv)$. It follows that:

$$\omega(P_H) \leq \varphi(s, p) \cdot \sum_{uv \in E(P_G)} \omega(uv) = \varphi(s, p) \cdot \omega(P_G) = \varphi(s, p) \cdot \omega_G^p(x, y)$$

as $\varphi(s, p) \geq 1$ and by definition of P_G .

Clearly, all edges of P_H are in H . Let us show now that P_H contains a p -multipath between x and y . We first assume x and y are non-adjacent in P_H . By Menger's Theorem applied between x and y in P_H since the removal of every set of at most $p - 1$ vertices in P_H does not disconnect x and y , then P_H has to contain a p -multipath between x and y .

Let S be any set of less than $p - 1$ faults in G . Since P_G is a p -multipath, P_G contains at least one path between x and y avoiding S . Let's call this path Q . For each edge uv of Q not in H , Q_{uv} is a p -multipath, so it contains one path avoiding S . Note that Q_{uv} may intersect Q_{wz} for different edges uv and wz of Q . If it is the case, then there is a path in $Q_{uv} \cup Q_{wz}$ from u to z (avoiding v and w), assuming that u, v, w, z are encountered in this order when traversing Q . Overall there must be a path connecting x to y and avoiding S in the subgraph $(Q \cap H) \cup \bigcup_{uv \in Q \setminus H} Q_{uv}$. By Menger's Theorem, P_H contains a p -multipath between x and y .

If x and y are adjacent in P_H , then we can subdivide the edge xy into the edges xz and zy by adding a new vertex z . Denote by P'_H this new subgraph. Clearly, if P'_H contains a p -multipath between x and y , then P_H too: a path using vertex z in P'_H necessarily uses the edges xz and zy . Now, P'_H contains a p -multipath by Menger's Theorem applied between x and y that are non-adjacent in P'_H .

We have therefore constructed a p -multipath between x and y in H of cost at most $\omega(P_H) \leq \varphi(s, p) \cdot \omega_G^p(x, y)$. It follows that $\omega_H^p(x, y) \leq \varphi(s, p) \cdot \omega_G^p(x, y)$ as claimed. \square

Theorem 5 is proved by applying Theorem 6 to the construction of Proposition 5, which is based on the distributed construction of s -hop spanners given by Proposition 4. Observe that the number of edges of the spanner is bounded by $O(kp^3 \cdot m(2n/p) \cdot \log n) = O(kp^{2-1/k}n^{1+1/k} \log^{2-1/k} n)$.

4.2.3 More faults lead to better stretch

In the previous section, we obtained through combinatorial means a result linking $p - 1$ fault tolerant spanners to vertex-disjoint p -multipath spanners. However the stretch was not satisfactory (from a $p - 1$ fault tolerant spanner of stretch s to a p multipath spanner of stretch $sp \cdot O(1 + p/s)^s$).

In Section 3.2.5, we showed that an edge-fault tolerant spanner tolerating more faults than $p - 1$ was also a edge-disjoint p -multipath spanner of the same stretch. In this section, we do a similar remark and show that by increasing the number of faults allowed by a vertex fault tolerant spanner we can have a vertex disjoint multipath spanner of the same stretch, thereby improving the result of Theorem 5. Namely, we show next that every vertex fault-tolerant s -spanner is a vertex multipath spanner with the same stretch, provided the number of faults is larger than the number of paths required.

Note that it is unclear how to generalize the analysis from Section 3.2.5 to vertex disjoint multipath spanners. To see this, recall that in Section 3.2.5 we considered an edge-disjoint p -path of minimal cost $\mathcal{S} = \{Q_1 \cup \dots \cup Q_p\}$ from x to y in G . We claim that every edge $e \in E(Q_i)$ does not contain too many bypasses that intersect with the other paths of \mathcal{S} . To prove this claim, we show that it is possible to substitute sub-paths of each Q_j , $j \neq i$ by a cheaper edge-disjoint-bypasses of the edge $e = (u, v)$ in the case that there are too many intersecting bypasses.

Suppose now that we want to translate this proof to vertex-disjoint multipaths. We now consider a vertex-disjoint p -path \mathcal{S}' of minimal cost between x and y . For every edge missing in the spanner, if we want to prove that it does not have too many bypasses intersecting with the other Q_j , we try to substitute sub-paths of each Q_j with some bypass of the edge $e = (u, v)$. The problem is that all these edge-disjoint bypasses are edge-disjoint but not vertex-disjoint. Specifically, all these bypasses contain the nodes u and v . Therefore, it is unclear how to use these bypasses to substitute multiple sub-paths and stay with vertex disjoint paths. We thus present a different analysis for vertex

disjoint multipath spanners at the price of slightly increasing the size of the spanner. Moreover, our analysis for vertex disjoint multipath spanners works only for unweighted graphs. We later show a simple construction for weighted vertex-disjoint p -multipath spanners with stretch sp (instead of s).

In this section we establish the following two theorems:

Theorem 7 *Given an unweighted graph G with n vertices, integer k and integer p , one can efficiently construct a vertex-disjoint p -multipath $(2k - 1)$ -spanner for G with $O\left(\phi(k, p) \cdot n^{1+\frac{1}{k}} \log n\right)$ edges, where $\phi(k, p) = (2k - 1)^{6\alpha} p^{2\alpha} + ((2k - 1) \cdot p)^{4\alpha}$ and $\alpha = 1 - \frac{1}{2k}$.*

Theorem 8 *Given a weighted graph G with n vertices, odd integer s and integer p , one can efficiently construct a vertex-disjoint p -multipath $(s \cdot p)$ -spanner for G with $O\left(\phi'(p, k) \cdot n^{1+\frac{1}{k}} \log n\right)$ edges, where $\phi'(p, k) = (p \cdot (2k - 1))^{2-\frac{1}{k}}$.*

The proof is a bit similar to the one of section 3.2.5. The idea of the proof is as follows. We start with any pair of nodes x, y , separated by an optimal (in terms of d^p) p -path in G , called $\mathcal{S} = \{Q_1 \cup \dots \cup Q_p\}$ in the rest. We consider an r -fault tolerant spanner H of G , for some large r . We take interest in the edges of this path which are not in the spanner. Contrary to the case of edge-fault tolerant spanners, absence of an edge (of G) in H does not guarantee the fact that there will be r vertex-disjoint paths in H . We determine the correct relationship between r and q the number of paths in Lemma 11. For an edge (belonging to some Q_i) missing in the spanner, Lemma 12 tells that there is only a small portion of the q bypasses that intersect with the other paths Q_j . Finally, lemmas 13 and 14 explain how to choose a bypass for each missing edge so that it does not intersect with other bypasses, with the help of Menger's theorem and a proof similar to the proof of Lemma 12.

A subgraph H is q -vertex-resilient with stretch s if for every edge $e = (x, y) \in E(G)$, either $e \in E(H)$ or H has at least q internal² vertex-disjoint s -paths between x and y . For a path P between two nodes x and y and a vertex $v \in V(P)$, let $\text{index}(v, P)$ be the distance (number of hops) between x and v in P . Two paths are said to intersect if they have at least one common vertex. A set of paths \mathcal{B} is called a set of V -bypasses (as shortname for vertex-disjoint bypasses) of u and v if the paths in \mathcal{B} are internal vertex-disjoint paths of length at most s between u and v . The next lemma shows that every vertex fault-tolerant s -spanner H has “many” V -bypasses between u and v for every edge $e = (u, v)$ in $E(G) \setminus E(H)$.

²that is except x and y

Lemma 11 *Every r -fault tolerant s -spanner is $\lfloor r/(s-1) \rfloor$ -vertex-resilient with stretch s .*

Proof. Consider an r -fault tolerant s -spanner H . Consider an edge $e = (u, v) \in E(G) \setminus E(H)$. We need to show that H contains $\lfloor r/(s-1) \rfloor$ V-bypasses between u and v such that $k < \lfloor r/(s-1) \rfloor$. Let \mathcal{B} be the set of these k V-bypasses. Note that $V(\mathcal{B}) \setminus \{u, v\}$ contains at most $(s-1) \cdot k < r$ vertices. Fix the set of vertices $F = V(\mathcal{B}) \setminus \{u, v\}$ to be faulty. Since the subgraph H is an r -fault tolerant s -spanner, by definition $H \setminus F$ contains an s -path between u and v . Therefore H contains more than k V-bypasses between u and v , contradiction. \square

Throughout, we consider a graph G . Consider two vertices x and y and let $\mathcal{S} = \{Q_1, \cup \dots \cup Q_p\}$ be a p vertex-disjoint multipath from x to y in G of minimal cost (i.e. whose cost is equal to $d^p(x, y)$). Consider an edge $e = (u, v)$ in one of the paths of \mathcal{S} . The next lemma shows that G does not contain “many” V-bypasses, namely, internal vertex disjoint paths of length at most s between u and v , that intersect with the other paths of \mathcal{S} .

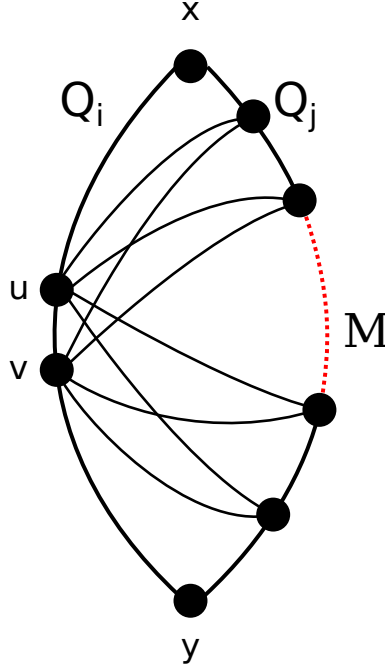
Lemma 12 *Consider two vertices x and y and let $\mathcal{S} = \{Q_1, \cup \dots \cup Q_p\}$ be a vertex-disjoint p -path from x to y in G of minimal cost. Consider an edge $e = (u, v) \in E(Q_i) \setminus E(H)$ for some $1 \leq i \leq p$. There are at most $2sp(p-1) + 2p(p-1)$ V-bypasses between u and v that intersect with $V(\mathcal{S}) \setminus V(Q_i)$.*

Proof. Assume by contradiction that there are more than $2sp(p-1) + 2p(p-1)$ such V-bypasses that intersect $V(\mathcal{S}) \setminus V(Q_i)$. By the Pigeonhole principle, there exists a path Q_j for some $j \neq i$ such that at least $2sp + 2p + 1$ of these V-bypasses intersect with Q_j . Let \mathcal{B} be the set of all these V-bypasses that intersect Q_j . For every V-bypass $A \in \mathcal{B}$, let $\mathbf{top}(A)$ be the earliest vertex of A on Q_j , i.e., the vertex in $V(A) \cap V(Q_j)$ with minimal $\mathbf{index}(\mathbf{top}(A), Q_j)$, and let $\mathbf{bottom}(A)$ be the last vertex of A on Q_j , i.e., the vertex in $V(A) \cap V(Q_j)$ with maximal $\mathbf{index}(\mathbf{top}(A), Q_j)$.

Let \mathcal{B}_h be the set of p V-bypasses $A \in \mathcal{B}$ with minimal $\mathbf{index}(\mathbf{top}(A), Q_j)$ and let \mathcal{B}_l be the set of p V-bypasses $A \in \mathcal{B}$ with maximal $\mathbf{index}(\mathbf{bottom}(A), Q_j)$. Let $A_h \in \mathcal{B}_h$ be the V-bypass with maximal $\mathbf{index}(\mathbf{top}(A_h), Q_j)$ and let $q_h = \mathbf{top}(A_h)$. Let $A_l \in \mathcal{B}_l$ be the V-bypass with minimal $\mathbf{index}(\mathbf{bottom}(A_l), Q_j)$ and let $q_l = \mathbf{bottom}(A_l)$. Let $M = Q_j[q_h, q_l]$ (i.e., the sub-path of Q_j from q_h to q_l). See Figure 4.4 for illustration.

We claim that:

- (1) the subgraph $H'(V, \mathcal{S}')$ for $\mathcal{S}' = (E(\mathcal{S}) \setminus E(M)) \cup E(\mathcal{B}_h) \cup E(\mathcal{B}_l)$ contains p vertex-disjoint paths from x to y ;

Figure 4.4: Illustration of the sets $\mathcal{B}_h, \mathcal{B}_l$ ($p = 2$) and the path M .

(2) $\omega(\mathcal{S}') \leq \omega(\mathcal{S})$.

To prove claim (1) we use Menger's theorem. We show that there is no set F of $p - 1$ vertices such that x and y are disconnected in $\mathcal{S}' \setminus F$. Consider a set F of at most $p - 1$ vertices. If F fails to intersect a path $Q_r \in \mathcal{S}$, for some $r \neq j$, then clearly x and y are connected in $\mathcal{S}' \setminus F$. So suppose the set F disconnects every path $Q_r \in \mathcal{S}$, for $r \neq j$, hence F contains exactly one vertex from each path $Q_r \in \mathcal{S}$ for every $r \neq j$. In particular, F contains only one vertex of Q_i . Therefore, one of u or v is not in F . Assume without loss of generality that $u \notin F$. Note that both sets \mathcal{B}_h and \mathcal{B}_l contain p V-bypasses. Since F contains at most $p - 1$ vertices, there must be a V-bypass $B_h \in \mathcal{B}_h$ and a V-bypass $B_l \in \mathcal{B}_l$ whose internal vertices are not in F . Let $x_1 = \mathbf{top}(B_h)$ and $y_1 = \mathbf{bottom}(B_l)$. Note that the sub-paths $Q_j[x, x_1]$ and $Q_j[y_1, y]$ do not contain any vertex from F , as $F \cap V(Q_j) = \emptyset$. Moreover, the V-bypasses B_h and B_l contain sub-paths $B_h[x_1, u]$ and $B_l[u, y_1]$ that do not intersect F . Concatenating all these paths together, we get a path $Q_j[x, x_1] \circ B_h[x_1, u] \circ B_l[u, y_1] \circ Q_j[y_1, y]$ from x to y . We thus conclude that \mathcal{H}' contains p vertex-disjoint paths from x to y , establishing (1). Next, we show claim (2). Recall that \mathcal{B} contains at least $2sp + 2p + 1$ V-bypasses intersecting Q_j . Moreover, each of the sub-paths $Q_j[x, q_h]$ and $Q_j[q_l, y]$ intersect with exactly p V-bypasses from \mathcal{B} . We get that the remaining part of Q_j , namely, the path $M = Q_j[q_h, q_l]$, intersects with at least $2sp + 2p + 1 - 2p = 2sp + 1$

V-bypasses from \mathcal{B} . Thus, the length of M is at least $2sp + 1$. In contrast, the number of edges in the edge collection $E(\mathcal{B}_h) \cup E(\mathcal{B}_l)$ that replaced M in \mathcal{S} is at most $2ps$. Hence, $\omega(\mathcal{S}') < \omega(\mathcal{S})$.

Finally parts (1) and (2) of the claim imply a contradiction to the optimality of \mathcal{S} . The lemma follows. \square

$$\text{Let } f = (4s + 2)(p - 1)s + 1 + 2sp(p - 1) + 2p(p - 1).$$

Lemma 13 *Every f -vertex-resilient subgraph H is a p -vertex disjoint multipath spanner.*

Proof. Consider two vertices x and y and let $\mathcal{S} = \{Q_1, \dots, Q_p\}$ be the p vertex-disjoint multipath from x to y in G of minimal cost. Consider an edge $e = (u, v) \in E(Q_i)$ such that $e \notin E(H)$ for some $1 \leq i \leq p$. By definition of H , H contains f V-bypasses between u to v . By Lemma 12, there are at most $2sp(p - 1) + 2p(p - 1)$ V-bypasses between u to v that intersect $V(\mathcal{S}) \setminus V(Q_i)$. We thus get that there exists a set **Bypasses**(e) of at least $f - 2sp(p - 1) - 2p(p - 1)$ V-bypasses between u to v in H , that do not intersect $V(\mathcal{S}) \setminus V(Q_i)$. We now show how to select for each edge $e \in Q_i$ for some i a V-bypass $B_e \in \mathbf{Bypasses}(e)$, such that B_e is vertex disjoint with any $B_{e'}$ for any $e' \in Q_j$ such that $e' \notin H$ and $j \neq i$.

Consider an edge $e = (u, v) \in Q_i$ such that $e \notin E(H)$. Let E_e be the set of edges e' such that $e' \in E(Q_j) \setminus E(H)$ and $V(B) \cap V(B') \neq \emptyset$ for some $j \neq i$, $B \in \mathbf{Bypasses}(e)$ and $B' \in \mathbf{Bypasses}(e')$. Towards proving Lemma 13, we first prove the next auxiliary lemma.

Lemma 14 *For every edge $e \in E(\mathcal{S}) \setminus E(H)$, the set E_e contains at most $(4s + 2)(p - 1)$ edges.*

Proof. Assume, towards contradiction, that $|E_e| \geq (4s + 2)(p - 1) + 1$. By the Pigeonhole principle, there is a path Q_j (for $j \neq i$) such that $|E(Q_j) \cap E_e| > 4s + 2$. Let $e_h = (u_h, v_h)$ be the edge in $E(Q_j) \cap E_e$ closest to x in Q_j , and let $e_l = (u_l, v_l)$ be the edge in $E(Q_j) \cap E_e$ closest to y in Q_j . Let $h_1, h_2 \in \mathbf{Bypasses}(e)$, $h_3 \in \mathbf{Bypasses}(e_h)$ and $h_4 \in \mathbf{Bypasses}(e_l)$ such that h_1 and h_3 intersect and h_2 and h_4 intersect (it could be that $h_1 = h_2$). Let M be the sub-path $Q_j[v_h, v_l]$.

We now claim that:

- (1) the subgraph $H''(V, \mathcal{S}'')$ for $\mathcal{S}'' = E(\mathcal{S}) \setminus E(M) \cup E(h_1) \cup E(h_2) \cup E(h_3) \cup E(h_4)$ contains p vertex disjoint paths from x to y ;
- (2) $\omega(\mathcal{S}'') < \omega(\mathcal{S})$.

We show claim (1) by using Menger's theorem to establish that \mathcal{S}'' contains p vertex-disjoint paths from x to y . Consider a set F of at most $p - 1$ vertices. If

F fails to intersect a path $Q_r \in \mathcal{S}$, for some $r \neq j$, then clearly x and y are connected in $\mathcal{S}'' \setminus F$. So suppose the set F disconnects every path $Q_r \in \mathcal{S}$, for $r \neq j$, hence F contains exactly one vertex from each path $Q_r \in \mathcal{S}$ for every $r \neq j$. Note that all vertices in h_3 and h_4 are not in F as h_3 and h_4 are disjoint from all Q_r for $r \neq j$ and we assume that F contains only nodes from Q_r for some $r \neq j$. Since F contains exactly one vertex in Q_i then one of u and v are not in F , assume w.l.o.g. that $u \notin F$. Let r_3 be a vertex in $h_3 \cap h_1$ and let r_4 be a vertex in $h_4 \cap h_2$. Note that the subgraph $(E(Q_j) \setminus E(M)) \cup E(h_3)$ contains a path from x to r_3 that does not intersect F . Similarly, the subgraph $(E(Q_j) \setminus E(M)) \cup E(h_4)$ contains a path from r_4 to y that does not intersect F . The cycle $h_1 \cup \{(u, v)\}$ contains at most one vertex in F and thus there is a path from r_3 to u that does not contain vertices from F . Similarly, $h_2 \cup \{(u, v)\}$ contains a path from r_4 to u that does not contain vertices from F . Concatenating all these paths together we get a path from x to y . Claim (1) follows. Next, we show claim (2) and thus derive a contradiction to the optimality of \mathcal{S} . The path Q_j contains at least $4s + 3$ edges from E_e and since the path M contains all these edges except $(e_h$ and $e_l)$, the length of M is at least $4s + 1$. In contrast, the number of edges in the edge set $E(h_1) \cup E(h_2) \cup E(h_3) \cup E(h_4)$ that replaced M in \mathcal{S} is at most $4s$. We thus get that $\omega(\mathcal{S}'') < \omega(\mathcal{S})$. This implies a contradiction to the optimality of \mathcal{S} . The lemma follows. \square

Consider the edges $e \in E(\mathcal{S}) \setminus E(H)$ one by one. For each edge $e \in E(Q_i) \setminus E(H)$, choose a V-bypass B_e that does not intersect with any $B_{e'}$ for an edge $e' \in E(Q_r) \setminus E(H)$ for some $r \neq i$. We claim that this process never gets stuck, namely, each time we consider an edge $e \in E(Q_i) \setminus E(H)$, there is at least one V-bypass in $\mathbf{Bypasses}(e)$ that does not intersect with the other V-bypasses selected so far. Let $\tilde{E}_e \subseteq E_e$ be the set of edges that were considered before e by this process. Note that $|\tilde{E}_e| \leq (4s + 2)(p - 1)$ by Lemma 14. Moreover, note that each path $B_{e'}$ for some $e' \in \tilde{E}_e$ intersects with at most s V-bypasses in $\mathbf{Bypasses}(e)$. Since there are more than $(4s + 2)(p - 1)s$ V-bypasses in $\mathbf{Bypasses}(e)$, at least one of these V-bypasses does not intersect with any of $B_{e'}$ for $e' \in \tilde{E}_e$. For each path $Q_r \in \mathcal{S}$ for $1 \leq r \leq p$, construct a path \tilde{Q}_r as follows. For every edge $e \in E(Q_r)$, if $e \in E(H)$ then add e to \tilde{Q}_r , otherwise add B_e to \tilde{Q}_r . It is not hard to verify that $V(Q_i) \cap V(Q_j) = \emptyset$ for any $i \neq j$ and that each \tilde{Q}_r is a path from x to y such that $\omega(\tilde{Q}_r) \leq s \cdot \omega(Q_r)$. The lemma follows. \square

The following theorem was shown by Dinitz and Krauthgamer in [DK11] (see also section 4.2.2.3).

Theorem 9 [DK11] *For every graph $G = (V, E)$, odd integer s and integer r , one can*

construct in polynomial time with high probability an r -vertex fault-tolerant s -spanner with $O\left(r^{2-\frac{2}{s+1}}n^{1+\frac{2}{s+1}}\log n\right)$ edges.

Combining Theorem 9, Lemma 11 and Lemma 13, we get Theorem 7.

For weighted graphs, we need the following supplementary lemma:

Lemma 15 *Every p -vertex-vertex-resilient subgraph H with stretch s is a p -vertex multipath $(s \cdot p)$ -spanner.*

Proof. To see this, consider two vertices x and y and let $\mathcal{S} = \{Q_1, \dots, Q_p\}$ be the set of p vertex-disjoint paths from x to y in G of minimal cost. We now show how to construct a subgraph $H'(V, \mathcal{S}')$ such that:

- (1) H' contains p vertex-disjoint paths from x to y ;
- (2) $\omega(\mathcal{S}') \leq s \cdot \omega(\mathcal{S})$.

Initially, set $\mathcal{S}' = \emptyset$. For every edge $e = (u, v) \in E(Q_r)$, if $e \in E(H)$ then add e to \mathcal{S}' , otherwise add to \mathcal{S}' a set $\mathcal{B}(e)$ of V -bypasses such that $|\mathcal{B}(e)| = p$ and $E(\mathcal{B}(e)) \subseteq E(H)$. Note that such a set $\mathcal{B}(e)$ exists since H is p -vertex-resilient with stretch s .

To prove claim (1) we use Menger's theorem. We show that there is no set F of $p - 1$ vertices such that x and y are disconnected in $\mathcal{S}' \setminus F$. Consider a set F of at most $p - 1$ vertices. Note that, for every edge $e \in (E(Q_r) \setminus E(H))$, there exists a V -bypass $B(e) \in \mathcal{B}(e)$ that is internal vertex disjoint from F . To see this, recall that $\mathcal{B}(e)$ contains p V -bypasses. In addition, there exists at least one path $Q_r \in \mathcal{S}$ such that $V(Q_r) \cap F = \emptyset$ for some $1 \leq r \leq p$. Construct the path Q'_r as follows. For every edge $e = (u, v) \in E(Q_r)$, if $e \in E(H)$ then add e to Q'_r , otherwise add the alternative path $B(e)$ to Q'_r . It is not hard to verify that Q'_r goes from x to y , $E(Q'_r) \subseteq E(H)$, and $V(Q'_r) \cap F = \emptyset$. Claim (1) follows.

To see claim (2), note that for every edge $e \in E(\mathcal{S})$, either e itself or an alternative path of length s is added to $E(\mathcal{S}')$. We get that $\omega(\mathcal{S}') \leq s \cdot \omega(\mathcal{S})$. \square

We note that Lemma 11 is also true, provided the spanner built is s -hop, as described in Section 4.2.2, especially Proposition 5. This in turn satisfies the property that every edge e is either included in the spanner H or H contains an alternative path connecting the end-vertices of e of length at most $s \cdot \omega(e)$ and with at most s hops.

Combining with Theorem 9, this allows us to conclude Theorem 8.

4.2.4 Towards an additive bi-path spanner

So far, none of the constructions presented earlier can yield less than stretch 3 spanners. Indeed, if we take the approach of Section 4.2.2, the best we can do with 2-paths and multiplicative spanners is $\varphi(3, 2) = 18$ by Theorem 5 and Theorem 6. Likewise, if we take the approach of Section 4.2.3, the best multiplicative stretch obtained will be the one of the underlying multiplicative fault-tolerant spanner, which is at least 3.

In this section, we describe an ad-hoc algorithm to construct a multipath spanner with a multiplicative stretch less than 3, with 2 paths, and aim to prove the following result:

Theorem 10 *Every weighted graph with n vertices and maximum edge weight W has a vertex-disjoint 2-multipath $(2, O(W))$ -spanner of size $O(n^{3/2})$ that can be constructed in $O(n^4)$ time.*

While the earlier constructions were based on derivation from fault-tolerant spanners, we use here a much more direct approach derived from classical spanner constructions. Furthermore, the spanners constructed are essentially additive.

As the 2-multipath distance $\overset{v}{d}^2(u, v)$ between u and v is essentially the shortest elementary cycle between u and v , we will focus on cycles in this section.

4.2.4.1 Construction

Classical spanner constructions make extensive use of balls, neighborhoods and trees. These structures are however unsuitable for the $\overset{v}{d}^2$ graph metric. For instance, if we define a ball of center u and radius r as the subgraph induced on all vertices v with $\overset{v}{d}^2(u, v) \leq r$, two vertices of the ball could be in different bi-connected components. We redefine next these structures.

Consider a weighted graph G and with an edge uv that is not a cut-edge³. Let $G[uv]$ denote the bi-connected component of G containing uv , and let $c_H^2(uv, w)$ be the minimum cost of a cycle in subgraph H passing through the edge uv and vertex w , if it exists and ∞ otherwise.

We define a *2-path spanning tree of root uv* as a minimal subgraph T of G such that every vertex w of $G[uv]$ belongs to a cycle of T containing uv . Such definition is motivated by the following important property (see Property 1 in paragraph 4.2.4.3): for all vertices a, b in $G[uv] \setminus \{u, v\}$, $\overset{v}{d}_G^2(a, b) \leq c_T^2(uv, a) + c_T^2(uv, b)$. This can be seen as a variant of the triangle inequality.

³A cut-edge is an edge that does not belong to a cycle.

If $c_T^2(uv, w) = c_G^2(uv, w)$ for every vertex w of $G[uv]$ then T is called a *shortest 2-path spanning tree*. An important point, proved in Lemma 16 in paragraph 4.2.4.2, is that such T always exists and contains $O(\nu)$ edges, ν being the number of vertices of $G[uv]$.

In the following we denote by $B_G^2(uv, r) = \{w : d_G^2(uv, w) \leq r\}$ and $B_G(u, r) = \{w : d_G(u, w) \leq r\}$ the 2-ball (resp. 1-ball) of G centered at edge uv (at vertex u) and of radius r . We denote by $N_G(u)$ the set of neighbors of u in G . We denote by $\text{BFS}(u, r)$ any shortest path spanning tree of root u and of depth r (not counting the edge weights). Finally, we denote by $\text{SPST}_G^2(uv)$ any shortest 2-path spanning tree of root uv in $G[uv]$.

The spanner H is constructed by Algorithm 2 for any weighted graph G having n vertices and maximum edge weight W . Essentially, the main loop of the algorithm selects an edge uv from the current graph lying at the center of a dense bi-component, adds to the spanner H a shortest 2-path spanning tree rooted at uv , and then destroys the neighborhood of uv .

```

 $F := G, H := (\emptyset, \emptyset);$ 
while  $\exists uv \in E(G), |B_G^2(uv, 4W) \cap (N_G(u) \cup N_G(v))| > \sqrt{n}$  do
   $H := H \cup \text{SPST}_F^2(uv) \cup \text{BFS}_G(u, 2) \cup \text{BFS}_G(v, 2);$ 
   $G := G \setminus (B_G^2(uv, 4W) \cap (N_G(u) \cup N_G(v)))$ 
 $H := H \cup G$ 

```

Algorithm 2: Construction of H .

4.2.4.2 Size analysis

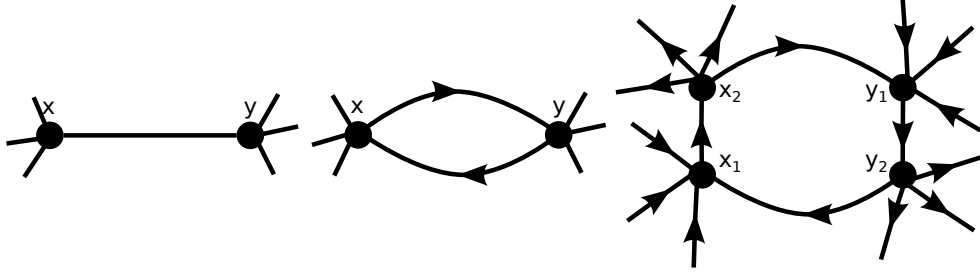
The proof of the spanner's size is done in two steps, thanks to the two next lemmas.

First, Lemma 16 shows that the while loop does not add too many edges: a shortest 2-path spanning tree with linear size always exists: it is built upon the algorithm of Suurballe-Tarjan [ST84] for finding shortest pairs of edge-disjoint paths in weighted digraphs.

Lemma 16 *For every weighted graph G and for every non cut-edge uv of G , there is a shortest 2-path spanning tree of root uv having $O(\nu)$ edges where ν is the number of vertices of $G[uv]$. It can be computed in time $O(n^2)$ where n is the number of vertices of G .*

Proof. In the following, we call $X = G[uv]$. $\text{SPST}_X^2(uv)$ will therefore be equal to $\text{SPST}_G^2(uv)$.

Let $\nu = |V(X)|$ and $\mu = |E(X)|$. We first show that we can reduce our problem to the problem of finding a one-to-all pair of edge-disjoint paths in a

Figure 4.5: Process by which X is transformed into X' .

directed graph. In other words, let \mathcal{P} be a procedure which yields a 2-(edge-disjoint)-tree rooted in a single vertex w in a directed graph X' . We show that we can derive \mathcal{P}' yielding $\text{SPST}^2_X(uv)$ from \mathcal{P} .

First, remark that the problem of finding $\text{SPST}^2_X(uv)$ is equivalent to finding the same structure but rooted in a single vertex w where the edge uv is replaced by uw, wv , and the weights of each edge uw and wv being equal to half of $\omega(uv)$.

We construct X' as follows: each undirected edge is replaced by two edges going in opposite direction and of the same weight. Then each vertex a is replaced by two vertices a_1 and a_2 where every incoming edge arrives at a_1 and every leaving edge leaves from a_2 . An edge going from a_1 to a_2 is finally added. Fig. 4.5 shows what happens to edges of X .

Note that $\nu' = |V(X')| = 2 \cdot (\nu + 1)$ and $\mu' = |E(X')| = 2 \cdot (\mu + 1) + \nu + 1$.

The procedure \mathcal{P}' proceeds as follows:

1. uv is replaced by uw, wv .
2. X' is constructed.
3. \mathcal{P} is called on X' , with the root vertex being w_2 .
4. Every edge of type $x_2 \rightarrow y_1$ present in the result of \mathcal{P} causes the addition of the edge xy to the result of \mathcal{P}' .

Two edge-disjoint paths in X' are vertex-disjoint in X . Indeed, as they cannot both use an edge of the type $x_1 \rightarrow x_2$ they cannot share x_1 or x_2 (except at the extremities) because the only way to leave (resp. arrive) from x_1 (resp. to x_2) is to use the edge $x_1 \rightarrow x_2$. So if we have two edge-disjoint paths in X' going from w_2 to some x_1 , the reduction back to X will yield two disjoint paths from w to x , and then from the edge wv to x .

We use the same procedure \mathcal{P} from Suurballe and Tarjan in [ST84] as in section 3.2.4. While not directly constructing the 2-(directed-edge-disjoint)-tree rooted in a single vertex w it can be extracted from their algorithm. This

procedure yields a structure with at most $2(\nu' - 1)$ edges. See the proof of Lemma 6 for a more extensive discussion.

Therefore the number of edges yielded by procedure \mathcal{P}' is at most $4 \cdot \nu$, which is $O(\nu)$. \square

Secondly, Lemma 17 shows that the graph G remaining after the while loop has only $O(n^{3/2})$ edges. For that, G is considered as an unweighted graph (edge weights are set to one) and we apply Lemma 17 with $k = 2$. The result we present is actually more general and interesting in its own right. Indeed, it gives an alternative proof of the well-known fact that graphs with no cycles of length $\leq 2k$ have $O(n^{1+1/k})$ edges since $B_G^2(uv, 2k) = \emptyset$ in that case.

Lemma 17 *Let G be an unweighted graph with n vertices, and $k \geq 1$ be an integer. If for every edge uv of G , $|B_G^2(uv, 2k) \cap N_G(u)| \leq n^{1/k}$, then G has at most $2 \cdot n^{1+1/k}$ edges.*

Proof. Consider Algorithm 3 applied to graph G . When the procedure terminates, all the vertices and edges of the graph have been removed. In the following, we count the number of edges removed by each step of the while loop, which in turn allows us to bound the total number of edges of G .

```

for  $i := k - 1$  to  $0$  do
    while  $\exists u, |B_G(u, i)| \geq n^{i/k}$  do
         $G := G \setminus B_G(u, i)$ 
    
```

Algorithm 3: Remove 1-balls.

Let X_i denote the set of vertices u whose 1-ball $B_G(u, i)$ is removed during iteration i of the **for** loop. Let $m(u)$ be the number of edges deleted when removing $B_G(u, i)$. Note that as $\sum_i \sum_{u \in X_i} |B_G(u, i)| = n$ (the procedure removed all the vertices), and that $\sum_i |X_i| \cdot n^{i/k} \leq n$ because each 1-ball is larger than $n^{i/k}$.

At each step, we argue that

$$m(u) \leq (n^{1/k} + 1) \cdot |B_G(u, i)| + |N_G(u, i + 1)|$$

where $N_G(u, i + 1)$ is the set of vertices at exactly $i + 1$ hops from u .

To this effect, let's consider a shortest path tree T rooted in u and spanning $B_G(u, i)$.

The number of edges in T is bounded by $|B_G(u, i + 1)|$, which can be decomposed in $|B_G(u, i)| + |N_G(u, i + 1)|$.

We can also bound the total number of non-tree edges as follows: for any $x \in B_G(u, i)$, let's consider $B_G^2(xy, 2k) \cap N_G(x)$, where y is the parent of x in T . We know that the number of vertices in this 2-ball is less than $n^{1/k}$ because it is a property of G . But $|B_G^2(xy, 2k) \cap N_G(x)|$ is also at least the number of non-tree edges attached to x : for an edge $xz \notin T$, the paths from z towards the root u and from x towards the root until they reach a common vertex are of length at most the radius of $B_G(u, i)$, which is $i \leq k$, and so there is a cycle of length at most $2k$ using the edges xz and xy . So the total number of non-tree edges is bounded by $n^{1/k} \cdot |B_G(u, i)|$.

The termination of the while loop during iteration $i + 1$ implies

$$|B_G(u, i + 1)| < n^{\frac{i+1}{k}}$$

or equivalently:

$$|N_G(u, i + 1)| < n^{\frac{i+1}{k}} - |B_G(u, i)|.$$

Therefore we have

$$m(u) < n^{1/k} |B_G(u, i)| + n^{\frac{1+i}{k}}.$$

And so

$$m(G) = \sum_{u \in \cup_i X_i} m(u) < n^{1/k} \sum_i \sum_{u \in X_i} |B_G(u, i)| + \sum_i |X_i| \cdot n^{\frac{i+1}{k}}$$

and as $\sum_i |X_i| \cdot n^{i/k} \leq n$, we have

$$|E(G)| \leq 2 \cdot n^{1+1/k}.$$

□

Combining these two lemmas we have:

Lemma 18 *Algorithm 2 creates a spanner of size $O(n^{3/2})$ in time $O(n^4)$.*

Proof. Each step of the while loop adds $O(n)$ edges from Lemma 16, and as it removes at least \sqrt{n} vertices from the graph this can continue at most \sqrt{n} times. In total the while loop adds $O(n^{3/2})$ edges to H .

After the while loop, the graph G is left with every $B_G^2(uv, 4W) \cap (N_G(u) \cup N_G(v))$ smaller than \sqrt{n} . If we change all edges weights to 1, it is obvious that every $B_G^2(uv, 4) \cap (N_G(u) \cup N_G(v))$ is also smaller than \sqrt{n} . Then as $B_G^2(uv, 4) \cap N_G(u)$ is always smaller than $B_G^2(uv, 4) \cap (N_G(u) \cup N_G(v))$, we can apply Lemma 17 for $k = 2$, and therefore bound the number of edges added in the last step of Algorithm 2.

The total number of edges of H is $O(n^{3/2})$.

The costly steps of the algorithm are the search of suitable edges uv and the cost of construction of SPST².

The search of suitable edges is bounded by the number of edges as an edge e which is not suitable can be discarded for the next search: removing edges from the graph cannot improve $B_G^2(e, 4W)$. Then for starting from one extremity of each edge a breadth first search of depth 3 must be computed, keeping only the vertices whose path in the search encounters the other extremity. The cost of the search is bounded by the number of edges of G . So in the end the search costs at most $O(n^4)$ steps.

The cost of building a SPST² is bounded by the running time of [ST84], which at worst costs $O(n^2)$ (the reduction is essentially in $O(m + n)$). Since the loop is executed at most \sqrt{n} times, the total cost is $O(n^{7/2})$.

So the total running time is $O(n^4)$. \square

4.2.4.3 Stretch analysis

The proof for the stretch is done as follows: we consider a, b two vertices such that $d_F^2(a, b) = \ell$ is finite (if it is infinite there is nothing to prove). We need to prove that the spanner construction is such that at the end, $d_H^2(a, b) \leq 2\ell + O(W)$. To this effect, we define $P_F = P_F^1 \cup P_F^2$ as a cycle composed of two disjoint paths (P_F^1 and P_F^2) going from a to b such that its weight sums to $d_F^2(a, b)$.

Proving the stretch amounts to show that there exists a cycle $P_H = P_H^1 \cup P_H^2$ joining a and b in the final H , with cost at most $2\ell + O(W)$. Observe that if the cycle P_F has all its edges in H then one candidate for P_H is P_F and we are done. If not, then there is at least one 2-ball whose deletion provokes actual deletion of edges from P_F (those are edges of P_F missing in the final H).

In the following, let uv be the root edge of the first 2-ball whose removal deletes edges from P_F (that is they are not added in H neither during the while loop nor the last step of the algorithm). Let G_i be the graph G just before the removal of $B_G^2(uv, 4W) \cap (N_G(u) \cup N_G(v))$, and G_{i+1} the one just after.

The rest of the discussion happens in the graph G_i otherwise noted.

The proof runs as follows: we first show in Lemma 19 that any endpoint of a deleted edge (of P_F) belongs to an elementary cycle comprising the edge uv and of cost at most $6W$. We then show in Lemma 20 that we can construct cycles using a and/or b passing through the edge uv , effectively bounding $c_H^2(uv, a)$ and $c_H^2(uv, b)$ due to the addition of the shortest 2-path spanning tree rooted at uv . Finally Lemma 21 shows that the union of a cycle passing through uv and a and another one passing through uv and b contains an elementary cycle joining

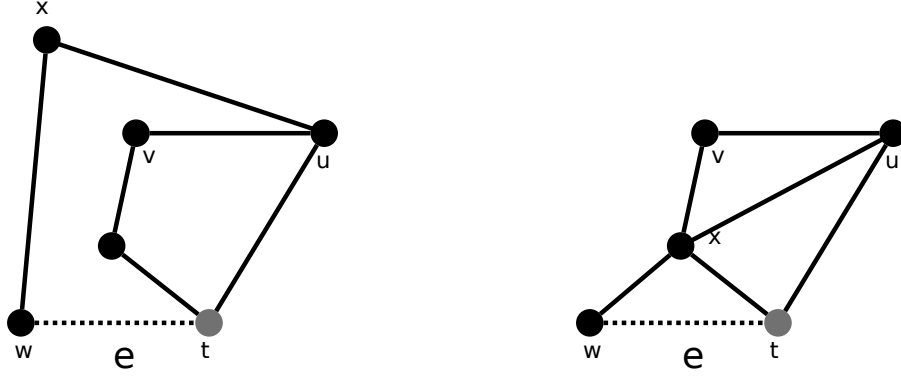


Figure 4.6: Proof of Lemma 19: a cycle of 6 hops exists between w and uv in G_i

a to b , its cost being at most the sums of the costs of the two original cycles.

Lemma 19 *Let $e = wt$ be an edge of $(E(G_i) \setminus E(G_{i+1})) \setminus E(H)$. Then in G_i both w and t are connected to uv by a cycle of cost at most $6W$.*

Proof. Since e is absent from both G_{i+1} and H , then at least one of its endpoints is in the vicinity of u or v in G_i . Without loss of generality we can consider t be a neighbour of u in G_i . Then w is at most two hops from uv in G_i .

We can first eliminate the case where w is a direct neighbor of v , as there is an obvious cycle of 4 hops: $w \rightarrow t \rightarrow u \rightarrow v \rightarrow w$.

Consider now the BFS tree rooted at u that is added to H . As w is at most two hops from u , there is a path $u \rightarrow x \rightarrow w$ in this tree (x is defined as the intermediate vertex of this path and may not exist). As e was removed, it means that x is distinct from t . Furthermore, t was removed because it belonged to the neighbourhood $B_G^2(uv, 4W)$ of some edge uv , so there is an elementary cycle of at most 4 hops passing through t and the edge uv .

Now we distinguish two cases as illustrated in Fig. 4.6.

If x is distinct (which is especially true when it does not exist) from an intermediate vertex between v and t in the cycle, then we can extract an elementary cycle of at most 6 hops passing through uv and w : $w \rightarrow x \rightarrow u \rightarrow v \rightarrow t \rightarrow w$. If x is the same as an intermediate vertex between v and t , then the cycle is $w \rightarrow t \rightarrow u \rightarrow v \rightarrow x \rightarrow w$. Hence we uncovered a cycle of length at most $6W$ containing both the edges e and uv .

□

We now show that we can use this lemma to exhibit cycles going from a to uv and from b to uv .

From the vertices belonging to both $B_{G_i}^2(uv, 6W)$ and P_F we choose the ones which are the closest to a and b (we know that at least two of them exist

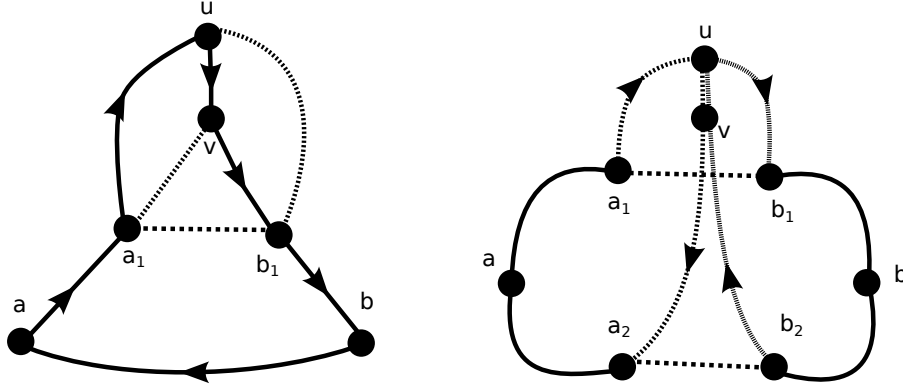


Figure 4.7: Proof of Lemma 21: the two cases for the simple paths.

because one edge was removed from P_F during step i of the loop). There are at maximum four of them (a_1, a_2, b_1, b_2) , one for each sub-path P_F^i and each extremity $\{a, b\}$. Note that each extremity is connected to the root edge by an elementary cycle of cost at most $6W$. Two cases are possible (the placement of the vertices can be seen on Fig. 4.7, although the paths on it are from the proof of Lemma 21):

Case 1: There are only two extremities (then they belong to the same sub-path) and their cycles which connect them to uv do not intersect the second sub-path (w.l.o.g we can suppose it is a_1 and b_1).

Case 2: There are more than two extremities: either some edges of the second path were removed or one of the cycles going from one of the extremities a_1 or b_1 to uv intersects the second path.

We show next that we can bound $c_H^2(uv, a)$ and $c_H^2(uv, b)$ with the help of the cycles connecting the endpoints and the path P_G . This is done with the two next lemmas.

Lemma 20 *For any two vertices w and t joined to the same edge uv by elementary cycles there is a simple path of cost at most the sum of the cycles' costs and passing through the edge uv .*

Proof. We show there is a simple path going from w to t passing through uv . Let us call Q^1 the elementary cycle joining w to uv and Q^2 the one joining t to uv . If going along the path Q^1 from w towards one of the endpoints of uv we do not encounter Q^2 , then the path from w to t is composed of the part of Q^1 , then the edge uv , then the part of Q^2 which reaches t without passing through uv . If it is not possible, then there are intersection points between Q^1 and Q^2 . Let i be the

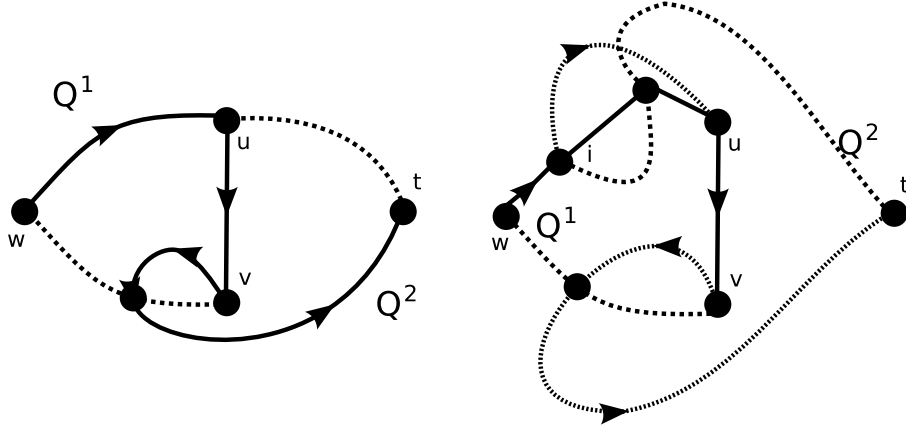


Figure 4.8: The two cases for Q^1 and Q^2 . The big dots represent the paths' unused portions.

closest intersection point from w . The path we are looking for is therefore $w \rightarrow i$ using Q^1 then $i \rightarrow uv \rightarrow t$ using the part of Q^2 which uses the edge uv (the other part would take us directly to a_2 without using uv). This path is simple because Q^2 is an elementary cycle and it cannot cross Q^1 before i because of the way i is chosen. The two cases are shown on figure 4.8.

□

Lemma 21 *Let a, b be two vertices of a graph G such that an elementary cycle of cost $d_G^2(a, b)$ has common vertices with some $B_G^2(uv, 6W)$. Then $c_G^2(uv, a)$ and $c_G^2(uv, b)$ are bounded by $d_G^2(a, b) + 12W$.*

Proof. The lemma is independent of the graph, but for clarity it will be proved using the graph G_i and P_F .

Recall that we distinguished two cases depending on whether $B_{G_i}^2(uv, 6W)$ intersects only one path of P_F (either P_F^1 or P_F^2) or both. Fig. 4.7 illustrates the proof of the two cases.

Lemma 20 allows us to solve the first case : since there are no intersections on the second path ($b \rightarrow a$), the cycle $a \rightarrow a_1 \rightarrow uv \rightarrow b_1 \rightarrow b \rightarrow a$ is simple. So there is a cycle in G_i joining uv, a and b of cost at most $12W + d_{G_i}^2(a, b)$. So $c_{G_i}^2(uv, a)$ is bounded by $12W + d_{G_i}^2(a, b)$ and so is $c_{G_i}^2(uv, b)$.

In the second case there are three or four extremities: a_1, b_1, a_2 , and b_2 , with possibly a_2 and b_2 being the same vertex. We then apply Lemma 20 twice: once for a_1 and a_2 and another time for b_1 and b_2 . That creates a simple cycle from a to uv passing via a_1 and a_2 and another one from b to uv passing via b_1 and b_2 .

These cycles are simple because the vertices were chosen to be the closest from a or b . Note that a_2 and b_2 can be the same. So

$$c_{G_i}^2(uv, a) \leq \omega(a \rightarrow a_1) + 12W + \omega(a_2 \rightarrow a) \leq d^2(a, b) + 12W$$

where $a \rightarrow a_1$ represents the arc between a and a_1 . The situation is the same for b . □

Property 1 *Let uv be a non cut-edge of G and T be any 2-path spanning tree rooted at uv . Then, for all vertices a, b in $G[uv] \setminus \{u, v\}$, $d_G^2(a, b) \leq c_T^2(uv, a) + c_T^2(uv, b) - \omega(uv)$.*

Proof. In T there is a cycle joining a to uv of cost $c_T^2(uv, a)$, and another one joining b to uv of cost $c_T^2(uv, b)$. Consider the subgraph P containing only the edges from these two cycles. The cost of P is $\omega(P) \leq c_T^2(uv, a) + c_T^2(uv, b) - \omega(uv)$ as edge uv is counted twice. It remains to show that P contains an elementary cycle between a and b . Note that since $a \notin \{u, v\}$, a has in P two vertex-disjoint paths leaving a and excluding edge uv : one is going to u and one to v . Similarly for vertex b .

Without loss of generality we can assume that a and b are not adjacent in P . Otherwise we can subdivide the edge ab to obtain a new subgraph P' . Clearly, if P' contains an elementary cycle between a and b , then P too. Consider that one vertex z , outside a and b , is removed in P . From the remark above, in $P \setminus \{z\}$, there must exist a path leaving a and joining some vertex $w_a \in \{u, v\} \setminus \{z\}$ and one path leaving b and joining some vertex $w_b \in \{u, v\} \setminus \{z\}$. If $w_a = w_b$, then a and b are connected in $P \setminus \{z\}$. If $w_a \neq w_b$, then edge uv belongs to $P \setminus \{z\}$ since in this case $z \notin \{u, v\}$, and thus a path connected a to b in $P \setminus \{z\}$. By Menger's Theorem, P contains a 2-multipath between a and b . □

Lemma 22 *H is a 2-multipath $(2, 24W)$ -spanner.*

Proof. If F contains a path of cost $d^2(a, b)$ with every edge in H , then there is nothing to prove. If there is some removed edge, then we can identify the iteration i which removed the first edge, and we can associate the graph G_i just before the deletion performed in the second step of the loop (so P_F is still intact in G_i). By virtue of Lemma 19 we can identify some root-edge uv and we know that there are some vertices of P_F linked to uv by an elementary cycle of length at most $6W$. Lemma 21 can then be applied, and so in G_i , $c_{G_i}^2(uv, a)$ and

$c^2_{G_i}(uv, b)$ are both bounded by $\overset{v}{d}^2_{G_i}(a, b) + 12W$. As the iteration's first step is to build a shortest 2-path spanning tree rooted in uv we know that in H

$$c^2_H(uv, a) \leq c^2_{G_i}(uv, a) \leq \overset{v}{d}^2_{G_i}(a, b) + 12W$$

and the same for b . Property 1 can then be used in the 2-path spanning tree, to bound $\overset{v}{d}^2_H(a, b)$:

$$\overset{v}{d}^2_H(a, b) \leq c^2_H(uv, a) + c^2_H(uv, b) \leq 2 \cdot \overset{v}{d}^2_{G_i}(a, b) + 24W$$

Finally, as in G_i P_F still exists completely, we have that $\overset{v}{d}^2_{G_i}(a, b) = \overset{v}{d}^2_F(a, b)$, so

$$\overset{v}{d}^2_H(a, b) \leq 2 \cdot \overset{v}{d}^2_F(a, b) + 24W$$

□

Chapter 5

Conclusion & perspectives

5.1 Review of the results

In this thesis, motivated by multipath routing considerations, we have introduced multipath spanners.

Spanners are a fundamental object linked to compact routing. Knowing that multipath routing is a requirement for many protocols, and after remarking that the theory of spanners could in theory be extended to any non decreasing graph metric, we considered vertex-disjoint and edge-disjoint multipath spanners.

The multipath metric comes into two versions, the edge-disjoint p -multipath graph metric d^p —which fulfills the distance axioms, in particular the triangle inequality—and the vertex-disjoint p -multipath graph metric v^p —which does not satisfy the triangle inequality. In this thesis we have studied these two types of multipath spanners which were consecutive to these two graph metrics.

5.1.1 Edge-disjoint multipath spanners

As edge-disjoint multipath spanners are built on a graph metric which is a proper distance, we studied these at first.

The first result we obtained was to note in Theorem 1 that a simple argument could be used to obtain p -multipath spanners for any p using $2k - 1$ -stretch classical spanners as building blocks, resulting in edge-disjoint p -multipath $p \cdot (2k - 1)$ -spanners with $O(p \cdot n^{1+1/k})$ edges, for any positive integers p, k . Unfortunately, although the number of edges is satisfactory—a mere p times the number of edges of a classical multiplicative spanner—the stretch is also multiplied by p compared to the stretch of the underlying spanners. Naively one could think that each path of a p -path would be stretched by the classical spanner's stretch factor, hence obtaining a p -multipath spanner of the same

stretch. This called for further examination.

We then showed in Proposition 2 that given a certain number of edges, an edge-disjoint p -multipath spanner cannot have a better stretch than an optimal 1-path spanner. In this light there seems to be a room for improvement from Theorem 1.

In this direction, we proved in Section 3.2.3 that in a very restricted setting—unweighted graph, 2-paths and stretch-3 underlying spanners—we could conduct a finer analysis and bring the stretch of Theorem 1 from 6 down to only 3, with the same number of edges.

To further improve the stretch, we showed in Section 3.2.4 that in the case of 2 paths we could reduce the multiplicative part of the stretch down to 2, while paying some additive factor. To this effect, we built in polynomial time an edge 2-multipath $(2, 8W)$ -spanner with $O(n^{3/2})$ edges, W being the maximum edge weight.

Finally we show in Section 3.2.5 that the fault-tolerant spanners devised by Peleg et al. in [CLPR10] for general graphs are related to multipath spanners. We noted that their algorithm for building p edge-fault-tolerant spanners is exactly the same as our building algorithm for p -multipath spanners. This leads to think that the two types of spanners are related. Indeed, we showed that for an edge f -fault-tolerant spanner to be considered as an edge-disjoint p -multipath spanner of the same stretch it required f to be a polynomial factor of p . More precisely, we show that a $kp^2 \log(W)$ -edge-fault-tolerant $2k - 1$ -spanner is also an edge-disjoint p -multipath $(2k - 1)$ -spanner. When using the algorithm of [CLPR10] for building f -edge-fault-tolerant spanners and using $(2k - 1)$ -spanners as a basis, the resulting spanner has $O(kp^2 \log(W)n^{1+1/k})$ edges.

This result shows a better stretch— $(2k - 1)$ —compared to Section 3.2.1— $p \cdot (2k - 1)$ —but with many more edges, thus making these two results incomparable.

5.1.2 Vertex-disjoint multipath spanners

After studying edge-disjoint spanners, we took interest in vertex-disjoint spanners. As noted by Remark 1, $\overset{v}{d}^p$ is not a proper distance function—for lack of triangle inequality. This complicates the design of new algorithms or adaptations of algorithms from classical spanner constructions to get vertex-disjoint multipath spanners. Therefore we studied these multipath spanners by examining their connection with fault-tolerant spanners.

In this direction, the first thing which we noted in Remark 2 was that a general weighted fault-tolerant spanner can have an arbitrary large stretch—linear in the number of vertices. However, we exhibit an additional property

which a fault-tolerant spanner can have in order to bound the multipath stretch: bounded hop spanners are such that a replacement path for an edge mustn't be composed of too many edges. When fault-tolerant spanners have this property we showed in Theorem 5 that it is possible to bound their stretch when considered as multipath spanners, albeit with some very large stretch. More precisely, we showed that we can get a vertex-disjoint p -multipath $kp \cdot O(1 + p/k)^{2k-1}$ -spanner with $O(kp^{2-1/k}n^{1+1/k}\log^{2-1/k}n)$ edges with the use of $p-1$ -fault-tolerant spanners from the works of Dinitz and Krauthgamer in [DK11].

In order to reduce this large stretch, we tried an idea similar to the one of Section 3.2.5: in order to decrease the multipath stretch down to the stretch of the multipath spanner, we were able to show in Section 4.2.3 that fault-tolerant $(2k-1)$ -spanners accepting a much larger number of faults than $p—O(s^3p + s^2p^2)—$ were in effect vertex-disjoint p -multipath spanners of the same stretch. However this comes with a larger number of edges— $O(\phi'(p, k) \cdot n^{1+\frac{1}{k}} \log n)$ edges, where $\phi'(p, k) = (p \cdot (2k-1))^{2-\frac{1}{k}}$ —compared to what we obtained in Section 4.2.2.

In another direction, we tried to build spanners with the smallest possible multiplicative stretch. Therefore in Section 4.2.4 we showed how to build a vertex-disjoint 2-multipath spanner of stretch $(2, O(W))—W$ being the maximum edge-weight—and with $O(n^{3/2})$ edges. The algorithm itself being an adaptation of the $(1, 2)$ stretch classical spanner construction. This algorithm needed some adaptation to the concept of balls in order to overcome the difficulty of losing the triangle inequality on which spanner algorithms implicitly rely on.

5.2 Perspectives

5.2.1 Improvements

Additive spanners. For each type of multipath spanners we managed to obtain spanners with an additive stretch in the case of 2 paths—see Sections 3.2.4 and 4.2.4. However we didn't obtain multipath spanners with less edges—say with an adaptation of the works of Baswana et al. in [BKMP10] where they obtain a $(1, 6)$ -spanner with $O(n^{4/3})$ edges—or with more paths. This calls for further research in this direction. For instance, is it possible to obtain vertex or edge p -multipath spanners with stretch $(\alpha, O(W))$ with $o(n^{3/2})$ edges and $\alpha < 2$, for $p > 1$?

Geometric multispanners. The present work was designed for general graphs. However spanners have been studied in some other contexts such as geometric graphs, where spanners are considered on complete graphs embedded in the euclidian plane. See [NS07] as an introduction to the topic.

We define the multipath distance in the same fashion as for general graphs. The original graph is seen as a complete weighted graph, and p -paths are simply collections of p vertex or edge disjoint paths between two vertices. A multipath spanner will be a spanning subgraph of this complete graph.

It's clear that our previous analysis for general graphs still apply. However we beleive that there exist size-stretch tradeoffs of much better quality—less edges for the same stretch, or a smaller stretch for less edges—than for general graphs for this multipath graph metric, as this is the case for $p = 1$ with Delaunay triangulations (cf. [Xia11]). For instance, can we have a constant stretch p -multipath spanner with $O(n)$ edges with $p > 1$?

Multipath routing. Another direction which was not examined at all in this thesis is the problem of multipath compact routing. Given a general graph, is there a way to transmit a certain number of packets in parallel while minimizing the treatment time and routing table size on each node ? The works of Suurballe and Tarjan in [ST84] where they build a one-to-all two-path structure may provide a starting point for computing distributed bi-path routing tables.

5.2.2 Non-decreasing graph metrics

In this thesis, we introduced two new possible extensions of spanners using non-decreasing graph metrics. In Chapter 3 we studied edge-disjoint multipath spanners which relied upon the edge-disjoint multipath graph metric. In Chapter 4 we studied vertex-disjoint multipath spanners built upon the vertex-disjoint multipath graph metric. In the two cases we showed that there existed a so-called size-stretch trade-off between the number of edges and the stretch.

We know that other non-decreasing graph metrics provide a size-stretch trade-off—besides the standard graph distance. For instance, in directed graphs the roundtrip distance provides such a trade-off, as shown by the works of Roditty et al. in [RTZ08]. In the same time they show that the one-way distance does not support such a trade-off. Note that this graph metric is not symmetric.

Moreover it is relatively to contruct graph metrics which do not sustain the trade-off, but nevertheless satisfy all the three axioms of distances—triangle inequality, symmetry, $d(x, y) = 0$ iff $x = y$. For instance, define the graph metric δ as follows:

$$\delta(u, v) = \begin{cases} 0 & \text{if } u = v \\ 1 & \text{if } \deg(u) \geq n/2 \text{ and } \deg(v) \geq n/2 \\ +\infty & \text{in every other case} \end{cases}$$

It is easy to check that this metric verifies the three axioms and is non-increasing.

On a complete bipartite graph the removal of one edge makes the distance between its two extremities go from 1 to $+\infty$. So there doesn't exist a trade-off between the number of edges and the stretch, despite δ being a proper distance.

This calls for an examination of criterias a given graph metric shall meet in order to sustain a size-stretch trade-off. This could be the subject of further works.

Bibliography

- [ACIM99] Donald Aingworth, Chandra Chekuri, Piotr Indyk, and Rajeev Motwani. Fast estimation of diameter and shortest paths (without matrix multiplication). *SIAM Journal on Computing*, 28(4):1167–1181, 1999.
- [ADD⁺93] Ingo Althöfer, Gautam Das, David Dobkin, Deborah A. Joseph, and José Soares. On sparse spanners of weighted graphs. *Discrete & Computational Geometry*, 9(1):81–100, 1993.
- [AHL02] Noga Alon, Shlomo Hoory, and Nathan Linial. The Moore bound for irregular graphs. *Graphs and Combinatorics*, 18(1):53–57, March 2002.
- [BKMP10] Surender Baswana, Telikepalli Kavitha, Kurt Mehlhorn, and Seth Pettie. Additive spanners and (α, β) -spanners, 2010. To appear.
- [BS09] Surender Baswana and Soumojit Sarkar. Fully dynamic randomized algorithms for graph spanners in centralized as well as distributed environments, March 2009.
- [CGP12] Shiri Chechik, Quentin Godfroy, and David Peleg. Multipath spanners via fault-tolerant spanners. In *The 1st Mediterranean Conference on Algorithms (MEDALG)*, December 2012.
- [CLPR10] Shiri Chechik, Michael Langberg, David Peleg, and Liam Roditty. Fault tolerant spanners for general graphs. *SIAM Journal on Computing*, 39(7):3403–3423, 2010.
- [CW00] Lenore Jennifer Cowen and Christopher Wagner. Compact roundtrip routing in directed networks. In *19th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 51–59. ACM Press, July 2000.

- [CZ04] Artur Czumaj and Hairong Zhao. Fault-tolerant geometric spanners. *Discrete and Computational Geometry*, 32(2):207–230, July 2004.
- [DGPV08] Bilel Derbel, Cyril Gavoille, David Peleg, and Laurent Viennot. On the locality of distributed sparse spanner construction. In *27th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 273–282. ACM Press, August 2008.
- [DK11] Michael Dinitz and Robert Krauthgamer. Fault-tolerant spanners: Better and simpler. Technical Report 1101.5753v1 [cs.DS], arXiv, January 2011.
- [Erd64] Paul Erdős. Extremal problems in graph theory. In *Publ. House Czechoslovak Acad. Sci., Prague*, pages 29–36, 1964.
- [ES82] Paul Erdős and Miklos Simonovits. Compactness results in extremal graph theory. *Combinatorica*, 2(3):275–288, 1982.
- [GGV10] Cyril Gavoille, Quentin Godfroy, and Laurent Viennot. Multipath spanners. In *17th International Colloquium on Structural Information & Communication Complexity (SIROCCO)*, volume 6058 of Lecture Notes in Computer Science, pages 211–223. Springer, June 2010.
- [GGV11] Cyril Gavoille, Quentin Godfroy, and Laurent Viennot. Node-disjoint multipath spanners and their relationship with fault-tolerant spanners. In *15th International Conference on Principles of Distributed Systems (OPODIS)*, volume 7109 of Lecture Notes in Computer Science, pages 143–158. Springer, December 2011.
- [JV09] Philippe Jacquet and Laurent Viennot. Remote spanners: what to know beyond neighbors. In *23rd IEEE International Parallel & Distributed Processing Symposium (IPDPS)*. IEEE Computer Society Press, May 2009.
- [KKKM07] Nate Kushman, Srikanth Kandula, Dina Katabi, and Bruce M. Maggs. R-bgp: Staying connected in a connected world. In *4th Symposium on Networked Systems Design and Implementation (NSDI)*, 2007.
- [Kle96] J.M. Kleinberg. *Approximation algorithms for disjoint paths problems*. PhD thesis, Massachusetts Institute of Technology, 1996.

- [Law76] Eugene L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, July 1976.
- [LG01] S.J. Lee and M. Gerla. Split multipath routing with maximally disjoint paths in ad hoc networks. In *IEEE International Conference on Communications (ICC)*, volume 10, pages 3201–3205, 2001.
- [LNL78] László Lovász, V. Neumann-Lara, and Michael D. Plummer. Mengerian theorems for paths of bounded length. *Periodica Mathematica Hungarica*, 9(4):269–276, 1978.
- [LNS98] Christos Levcopoulos, Giri Narasimhan, and Michiel Smid. Efficient algorithms for constructing fault-tolerant geometric spanners. In *30th Annual ACM Symposium on Theory of Computing (STOC)*, pages 186–195. ACM Press, May 1998.
- [MTG03] Stephen Mueller, Rose P. Tsang, and Dipak Ghosal. Multipath routing in mobile ad hoc networks: Issues and challenges. In *Performance Tools and Applications to Networked Systems, Revised Tutorial Lectures [from MASCOTS 2003]*, pages 209–234, 2003.
- [NCD01] Asis Nasipuri, Robert Castañeda, and Samir Ranjan Das. Performance of multipath routing for on-demand protocols in mobile ad hoc networks. *Mobile Networks and Applications*, 6(4):339–349, 2001.
- [NS07] Giri Narasimhan and Michiel Smid. *Geometric Spanner Networks*. Cambridge University Press, 2007.
- [Pet07] Seth Pettie. Low distortion spanners. In *34th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 4596 of Lecture Notes in Computer Science, pages 78–89. Springer, July 2007.
- [PS89] David Peleg and Alejandro A. Schäffer. Graph spanners. *J. of Graph Theory*, 13(1):99–116, 1989.
- [PSA05] P. Pan, G. Swallow, and A. Atlas. Fast Reroute Extensions to RSVP-TE for LSP Tunnels. RFC 4090 (Proposed Standard), 2005.
- [PT93] László Pyber and Zsolt Tuza. Menger-type theorems with restrictions on path lengths. *Discrete Mathematics*, 120(1-3):161–174, September 1993.
- [PU89] David Peleg and Jeffrey D. Ullman. An optimal synchronizer for the hypercube. *SIAM J. on Computing*, 18(4):740–747, 1989.

- [RTZ08] Liam Roditty, Mikkel Thorup, and Uri Zwick. Roundtrip spanners and roundtrip routing in directed graphs. *ACM Transactions on Algorithms*, 3(4):Article 29, June 2008.
- [ST84] J. W. Suurballe and Robert Endre Tarjan. A quick method for finding shortest pairs of disjoint paths. *Networks*, 14(2):325–336, 1984.
- [TZ05] Mikkel Thorup and Uri Zwick. Approximate distance oracles. *Journal of the ACM*, 52(1):1–24, January 2005.
- [Wen91] Rephael Wenger. Extremal graphs with no C^4 's, C^6 's, or C^{10} 's. *Journal of Combinatorial Theory, Series B*, 52(1):113–116, 1991.
- [Xia11] Ge Xia. Improved upper bound on the stretch factor of delaunay triangulations. In *27th Annual ACM Symposium on Computational Geometry (SoCG)*, pages 264–273. ACM-SIAM, June 2011.