A spanner of a graph is a subgraph that roughly preserves the pairwise distances of the graph. The benefit of spanners is to compress information about distances in a graph by looking at distances within a subgraph.

**Definition 0.1.** An \((\alpha, \beta)\)-spanner of \(G = (V, E)\) is a subgraph \(H = (V, E_H)\), \(E_H \subseteq E\), such that \(\forall u, v \in V\),

\[ d(u, v) \leq d_H(u, v) \leq \alpha d(u, v) + \beta. \]

The \(d(u, v) \leq d_H(u, v)\) is always true, since \(H\) is a subgraph of \(G\).

If \(\beta = 0\), then it’s called an \(\alpha\)-multiplicative spanner; if \(\alpha = 1\), then it’s called a \(+\beta\)-additive spanner; otherwise, it’s called a mixed \((\alpha, \beta)\)-spanner. In general, directed graphs don’t contain sparse spanners. An example is shown in Figure 1, where \(G\) is a directed bipartite graph, with all its edges leaving nodes in set \(U\) and pointing to nodes in set \(V\). In this case, any spanner with finite \((\alpha, \beta)\) must contain all edges in \(G\). As a result, directed graphs don’t always have sparse spanners. Thus, we will focus on spanners of undirected graphs in this class.

![Figure 1: An example of a directed graph which does not have a good spanner.](image)

Most of this lecture will be about additive spanners. At the end of this lecture, we will talk a bit about multiplicative spanners.

### 1 Additive Spanners

We restrict ourselves to unweighted graphs, in which an additive spanner makes sense. We will cover \(+2\), \(+4\) and \(+6\) spanners in this lecture.

**Theorem 1.1.** Any \(n\)-node graph \(G\) has a \(+2\)-spanner with \(O\left(n^{3/2} \log n\right)\) edges.

**Theorem 1.2.** Any \(n\)-node graph \(G\) has a \(+4\)-spanner with \(\widetilde{O}\left(n^{3/2}\right)\) edges.
**Theorem 1.3.** Any $n$-node graph $G$ has a $+6$-spanner with $\tilde{O}\left(n^{4}\right)$ edges.

Theorem 1.3 is optimum for spanners with constant additive errors. Due to the following theorem, any spanner with much fewer edges must have polynomial error.

**Theorem 1.4.** [1] There exist graphs with $n$ nodes such that if a spanner has $O(n^{4/3-\epsilon})$ edges for some $\epsilon > 0$, then it has an additive error at least $n^{\delta}$ for some $\delta > 0$.

**Remark 1.** It is an open problem whether we can get $+4$-additive spanners with $\tilde{O}(n^{4/3})$ edges.

**Proof of Theorem 1.1.** The proof of this theorem is very similar to the $+2$-approximation to the APSP problem covered in Lecture 13. Let $S$ be a hitting set for $\{N(v) \mid \deg(v) \geq \sqrt{n}\}$. Do a BFS search from each $s \in S$, and add the BFS tree rooted at $s$ to $E_H$. For every $u \in V$ with $\deg(u) < \sqrt{n}$, add all edges incident to $u$ to $E_H$. By construction, $E_H = O(|S| \cdot n) + O(n\sqrt{n}) = O\left(n^{\frac{2}{3}} \log n\right)$. Consider any pair of edge $(u, v) \in V$ with shortest path $P$ in $G$. We have two cases:

- $P$ contains only low-degree nodes. Then $P$ is entirely contained in $E_H$, so $d_H(u, v) = d(u, v)$.
- $P$ contains a high-degree node $x$. Let $s_x \in S$ be a node adjacent to $x$. Then we can approximate the distance from $u$ to $v$ by appending the paths from $u$ to $s_x$ and from $s_x$ to $v$, since $E_H$ contains shortest paths from $s_x$ to every other vertex. Thus
  $$d_H(u, v) \leq d_H(u, s_x) + d_H(s_x, v) = d(u, s_x) + d(s_x, v) \leq (d(u, x) + 1) + (d(x, v) + 1) = d(u, v) + 2.$$  

Therefore the $H$ constructed by this algorithm is a $+2$-spanner, as desired. □

The $+4$ and $+6$-additive spanners both rely on the following idea.

**Claim 1.** Let $P$ be a shortest path in $G = (V, E)$. Let $x \in V$, then $x$ has at most 3 neighbors on $P$.

**Proof.** Suppose for the sake of contradiction, $x$ has four neighbors on $P$, consecutively labeled as $v_1, v_2, v_3, v_4$. Since $v_1, v_2, v_3, v_4$ are vertices on a shortest path, the distance between $v_1$ and $v_4$ is at least 3. However, since $v_1$ and $v_4$ are both neighbors of $x$, their distance is at most 2, a contradiction. □

Using Claim 1, we can show the following Corollary, which will be used in the construction of both the $+4$ and $+6$ additive spanners.

**Corollary 1.1.** If a shortest path $P$ has at least $L$ nodes of degree at least $D$ for some $D \geq 4$, then there exists $\Omega(LD)$ distinct neighbors of the path $P$.

**Proof.** For each node $v \in P$ that has degree at least $D$, it has at most 3 neighbors in $P$ by Claim 1. Thus, each $v$ has at least $D - 3$ neighbors outside of $P$. For each neighbor of $P$, it can have at most 3 neighbors in $P$ by Claim 1, so the number of distinct neighbors of $P$ is at least $\frac{1}{2}(D - 3)L = \Omega(LD)$ when $D \geq 4$. □

Now we are ready to show the $+4$-additive spanner.

**Proof of Theorem 1.2.** This proof actually won’t be an algorithm; instead, it is a proof for the existence of the spanner. Let $D, L$ be two parameters of the algorithm. We will call a vertex with degree at least $D$ a “high degree” vertex and a vertex with degree less than $D$ a “low degree” vertex. For any pair of vertices $u, v$, we fix one arbitrary shortest path $P(u, v)$ between them.

Let $S \subseteq V$ where $|S| = O\left(\frac{n}{D^2} \log n\right)$ be a subset of vertices that hits the neighborhood of every high degree vertex. By Corollary 1.1, the size of the neighborhood of every $P(u, v)$ where $P(u, v)$ contains at least $L$ high degree nodes is $\Omega(DL)$. Thus, we can find $T \subseteq V$ where $|T| = O\left(\frac{n}{DL} \log n\right)$ that hits the neighborhood of every $P(u, v)$ where $P(u, v)$ contains at least $L$ high degree nodes.

Initially, let the edge set of $H$ be empty. The construction for a $+4$-spanner $H$ from $G$ is as follows:
1. For all low degree \( v \in V \), add all the edges incident to \( v \) to \( H \). This contributes at most \( n \cdot D = O(nD) \) edges.

2. For each \( s \in T \), add the entire breadth-first-search tree rooted at \( s \) into \( H \). Here, we add \( O(n |T|) = \tilde{O}(\frac{n^2}{DE}) \) edges.

3. For each high degree node \( x \), let \( s_x \) be one of its neighbors in \( S \), add \((x, s_x)\) to \( H \). We only add \( O(n) \) edges in this part.

4. For each \( s \in S \), we use \( N(s) \) to denote the set of neighbors of \( s \). We continue adding edges to \( H \) with Algorithm 1.

**Algorithm 1:** Adding final edges to \( H 

```
foreach distinct \( s, s' \in S \) do
    \( P_{s,s'} = \{\}\);  
    foreach \( a \in N(s), b \in N(s') \) do
        if some shortest path \( P \) from \( a \rightarrow b \) has at most \( L \) high degree nodes then
            \[ P_{s,s'} . insert(\{\{s,a\}\} \cup P \cup \{\{b,s'\}\}) ; \]
        if \( P = \{\}\) then
            continue;
        \( p = \) shortest path in \( P_{s,s'} \);
        foreach edge \( e \in p \) do
            \( H . insert(e) \);
```

The path added to \( H \) corresponding to \( s \) and \( s' \) will be referred to as the \((s, s')\)-linking path. There are \( \tilde{O}(\frac{n^2}{D^2}) \) pairs of \( s, s' \). For each pair, we may add the edges from some path \( P \) connecting \( s \) and \( s' \). The only edges in \( P \) not already in \( H \) are those between two high degree nodes and the edges \((s, a)\) and \((b, s')\), of which there are \( \leq L + 2 \) in \( P \). Thus each pair \( s, s' \) adds \( O(L) \) edges to \( H \). Summing over the pairs, \( \tilde{O}(\frac{n^2}{D^2} L) \) edges are added.

Summing over all steps, the number of edges in \( H \) is \( \tilde{O}(nD + \frac{n^2}{DL} + \frac{n^2}{DP} L) \). By setting \( D = n^{2/5} \) and \( L = n^{1/5} \), we get \( \tilde{O}(n^{7/5}) \) edges as promised. It remains to show that \( H \) is an additive 4-spanner of \( G \).

If a pair of vertices \( u \) and \( v \) in \( G \) have a shortest path using no high degree nodes, then that path is in \( H \) due to Step (1).

If \( P(u, v) \) contains at least \( L \) high degree vertices, then \( T \) hits a neighbor of the path \( P(u, v) \). Thus the edges added in Step (2) include a \(+2\)-approximation for a shortest path between such \( u \) and \( v \).

The only remaining case is \( uv \)-shortest paths hitting between 1 and \( L \) high degree nodes.

**Claim 2.** If \( P(u, v) \) contains between 1 and \( L \) high degree nodes, then after Step (4), \( d_H(u, v) \leq d(u, v) + 4 \).

**Proof.** Let \( x \) be the first and \( y \) be the last high degree nodes in \( P(u, v) \) (possibly not distinct). Recall that \( s_x, s_y \in S \) and we added edges \((x, s_x), (y, s_y)\) in Step (3). Let \( a \) and \( b \) be the neighbors of \( s_x \) and \( s_y \) respectively connected by the \((s_x, s_y)\)-linking path. Note that the \((s_x, s_y)\)-linking path exists because there is at least one pair of elements of \( N(s_x) \) and \( N(s_y) \) connected by a shortest path using \( \leq L \) high degree nodes, namely \( x \) and \( y \).

Since the subpath of \( P(u, v) \) from \( u \) to \( x \) and from \( y \) to \( v \) is in \( H \) (Step (1)), we have

\[
d_H(u, v) \leq d(u, x) + d_H(x, y) + d(y, v).
\]

Thus we need only show that \( d_H(x, y) \leq d(x, y) + 4 \).
Recall that \( a \) and \( b \) have the shortest path between them in \( G \) of any pair of elements in \( N(s_x) \) and \( N(s_y) \), excluding paths with greater than \( L \) high degree nodes. Namely, since \( x \in N(s_x) \) and \( y \in N(s_y) \), we have that \( d(a, b) \leq d(x, y) \). Plugging this in yields
\[
d_H(x, y) \leq d_H(x, s_x) + d_H(s_x, a) + d_H(a, b) + d_H(b, s_y) + d_H(s_y, y) = 1 + 1 + d(a, b) + 1 + 1 \leq d(x, y) + 4,
\]
completing the proof.

The construction for 6-additive spanner shares many ideas with the construction for the 4-additive spanner, so for simplicity in the proof we skip the proof of some claims that are already proved in the proof for Theorem 1.2.

**Proof of Theorem 1.3.** Let \( D \) be a parameter of the algorithm. We will call a vertex with degree at least \( D \) a “high degree” vertex and a vertex with degree less than \( D \) a “low degree” vertex. For any pair of vertices \( u, v \), we fix one arbitrary shortest path \( P(u, v) \) between them.

Let \( S \subseteq V \) where \( |S| = O(n/ \log n) \) be a subset of vertices that hits the neighborhood of every high degree vertex. Fix some \( 0 \leq j \leq \log n \). By Corollary 1.1, the size of the neighborhood of every \( P(u, v) \) where \( P(u, v) \) contains at least \( 2^{j+1} \) high degree nodes is \( \Omega(D \cdot 2^j) \). Thus, we can find \( S_j \subseteq V \) where \( |S_j| = O(n/ \log n) \) that hits the neighborhood of every \( P(u, v) \) that contains at least \( 2^j \) high degree nodes.

Initially, let the edge set of \( H \) be empty. The construction for a \( +6 \)-spanner \( H \) from \( G \) is as follows:

1. For all low degree \( v \in V \), add all the edges incident to \( v \) to \( H \). This contributes at most \( n \cdot D = O(nD) \) edges.
2. For each high degree node \( x \), let \( s_x \in S \) be one arbitrary neighbor of \( x \) in \( S \). We add \((x, s_x)\) to \( H \). We only add \( O(n) \) edges in this part.
3. We use \( N(v) \) to denote the neighborhood of vertex \( v \). We continue adding edges to \( H \) with Algorithm 2 for every integer \( j \in [0, \log n] \).

**Algorithm 2: Edge-Adding(\( j \))**

\[
\text{foreach } s \in S, s' \in S_j \text{ do}
\begin{align*}
P_{s, s'} &= \{\}; \\
\text{foreach } a \in N(s), b \in N(s') \text{ do}
\begin{align*}
&\text{if some shortest path } P \text{ from } a \rightarrow b \text{ has at most } 2^{j+1} \text{ high degree nodes then} \\
&P_{s, s'} \text{.insert}((\{s, a\} \cup P \cup \{b, s'\})); \\
&\text{if } P = \{\} \text{ then} \\
&\text{continue;}
\end{align*}
\end{align*}
\begin{align*}
p &= \text{shortest path in } P_{s, s''}; \\
\text{foreach edge } e \in p \text{ do}
\begin{align*}
&H \text{.insert}(e);
\end{align*}
\end{align*}
\]

The path added to \( H \) corresponding to \( s \) and \( s' \) during Edge-Adding(\( j \)) will be referred to as the \( j \)-th \((s, s')\)linking path. The total number of edges we added is \( O(\sum_{j=0}^{\log n} |S||S_j| \cdot 2^{j+1}) \). By plugging in \( |S| = \tilde{O}(n/D) \) and \( |S_j| = \tilde{O}(n/D^2) \), we get
\[
O \left(\sum_{j=0}^{\log n} |S||S_j| \cdot 2^{j+1}\right) = \tilde{O} \left(\sum_{j=0}^{\log n} \frac{n}{D} \cdot \frac{n}{D} \cdot 2^{j+1}\right) = \tilde{O} \left(\frac{n^2}{D^2}\right).
\]
Summing over the three steps, the number of edges in $H$ is $\tilde{O}(nD + \frac{n^2}{2\pi})$. By setting $D = n^{1/3}$, the number of edges in $H$ becomes $\tilde{O}(n^{4/3})$ as promised. It remains to show that $H$ is an additive $+6$-spanner of $G$.

If $P(u, v)$ does not contain any high degree vertex, then the edges added in Step (1) already contain $P(u, v)$, and thus $d_H(u, v) = d(u, v)$.

Now suppose there are $h$ high degree nodes on $P(u, v)$ for some $h \geq 1$. Pick $j$ such that $2^j \leq h < 2^{j+1}$. Let $x$ be the first high degree vertex on $P(u, v)$, $y$ be the last high degree vertex on $P(u, v)$. Since the path from $x$ to $y$ contains $h \geq 2^j$ high degree vertices, by the construction of $S_j$, there exists a vertex $s \in S_j$ that hits a neighbor on the shortest path from $x$ to $y$. Let $z \in P(x, y)$ be a neighbor of $s$. Recall we added an edge between $x$ and $s_x \in S_j$, and an edge between $y$ and $s_y \in S$ in Step (2).

Consider the following path in $H$ from $u$ to $v$. First, we take the path from $u$ to $x$ on $P(u, v)$ (edges added in Step (1)), then move from $x$ to $s_x$. From $s_x$ to $s$, we use the $j$-th $(s_x, s)$ linking path. We can show that the length of the $j$-th $(s_x, s)$ linking path is at most $d(x, s) + 2$, since the path $s_x \rightarrow x \rightsquigarrow z \rightarrow s$ is a valid candidate for $j$-th $(s_x, s)$ linking path. From $s$ to $s_y$, we use the $j$-th $(s_y, s)$ linking path, which has length at most $d(s, s_y) + 2$. We then take the edge from $s_y$ to $y$, and finally, take the path from $y$ to $v$ on $P(u, v)$. All edges on the path above are added to $H$, and the length of the path is at most $d(u, v) + 6$.

Therefore, $H$ is a $+6$-additive spanner.

\[\square\]

## 2 Multiplicative Spanners

In this section, we study multiplicative spanners. For simplicity, we use $\alpha$-spanner to refer to $\alpha$-multiplicative spanner in this section.

**Theorem 2.1.** Let $k \geq 1$ be an integer, then every $n$-node undirected weighted graph $G$ contains a $(2k-1)$-spanner with $O \left( n^{1+\frac{1}{k}} \right)$ edges.

Theorem 2.1 is tight if we assume the following popular conjecture known to be true for small values of $k$.

**Conjecture 1.** (Erdős girth conjecture) For integer $k \geq 1$ and sufficiently large $n$, there exist $n$-node undirected unweighted graphs of girth $\geq 2k + 2$ with $\Omega \left( n^{1+\frac{1}{k}} \right)$ edges.

**Claim 3.** The Erdős girth conjecture implies that the bound in Theorem 2.1 is tight, i.e. there exists some graph $G$ on $n$ nodes such that any $(2k-1)$-spanner has $\Omega \left( n^{1+\frac{1}{k}} \right)$ edges.

**Proof.** Let $G$ be an unweighted graph on $n$ edges with girth $2k + 2$ and $\Omega \left( n^{1+\frac{1}{k}} \right)$ edges, given by the Erdős girth conjecture. We’ll show that $G$ has no non-trivial $(2k-1)$-spanners.

Assume there exists some subgraph $H \subseteq G$ that is a $(2k-1)$-spanner for $G$. Choose some edge $(u, v) \in E - E_H$. By the definition of a spanner, $d_H(u, v) \leq (2k-1)d(u, v) = 2k - 1$. Therefore there exists some path $P$ in $E_H$ connecting $u, v$ with length at most $2k - 1$. However, adding $(u, v)$ to $P$ then completes a cycle in $G$ of length at most $2k$; since $G$ has girth at least $2k + 2$, this is a contradiction. \[\square\]

Now we prove Theorem 2.1.

**Proof of Theorem 2.1.** We can generate a $(2k-1)$-spanner using the Create-Spanner algorithm. We prove the correctness of this algorithm with the following three claims.

**Claim 4.** $H$ is a $(2k-1)$-spanner, i.e., $\forall u, v \in V, d_H(u, v) \leq (2k-1)d(u, v)$.

**Claim 5.** $H$ has girth greater than $2k$.

**Claim 6.** Any $n$-node graph with girth greater than $2k$ has $O \left( n^{1+\frac{1}{k}} \right)$ edges.
Algorithm 3: Create-Spanner($G$)

\[
E_H \leftarrow \emptyset.
\]

\[
\text{foreach } (u, v) \in E \text{ in non-decreasing weight order do}
\]

\[
\begin{aligned}
\text{if } d_H(u, v) > (2k - 1)w(u, v) & \text{ then} \\
E_H & \leftarrow E_H \cup (u, v)
\end{aligned}
\]

Return $H$.

Proof of Claim 4. Let $u, v$ be vertices in $V$, and $P$ be their shortest path in $G$. For each edge $(x, y)$ in $P$, either:

- $(x, y) \in E_H$
- There is some path in $H$ between $x, y$ of length at most $(2k - 1)w(x, y)$. If no such path exists, then $(x, y)$ would have been added to $E_H$ in Create-Spanner when it was considered.

Therefore

\[
d_H(u, v) \leq \sum_{(x, y) \in P} d_H(x, y) \leq \sum_{(x, y) \in P} (2k - 1)w(x, y) = (2k - 1)w(P) = (2k - 1)d(u, v).
\]

□

Proof of Claim 5. Assume $H$ has a cycle $C$ of length $\leq 2k$ for contradiction. Let $(u, v)$ be the edge of $C$ with largest weight and $(u, v)$ is the last edge in $C$ added to $E_H$. Thus, we must have

\[
\sum_{(x, y) \in C, (x, y) \neq (u, v)} w(x, y) > (2k - 1)w(u, v),
\]

since otherwise we wouldn’t add $(u, v)$ to $H$. On the other hand, each edge in the path $C \setminus \{(u, v)\}$ has weight at most $w(u, v)$ and there are $2k - 1$ edges on $C \setminus \{(u, v)\}$, so

\[
\sum_{(x, y) \in C, (x, y) \neq (u, v)} w(x, y) \leq (2k - 1)w(u, v).
\]

Thus, we have a contradiction. □

Proof of Claim 6. For the sake of contradiction, let $H$ be any graph with girth greater than $2k$ and at least $10n^{\frac{1}{k}}$ edges. Modify the graph by repeatedly removing any nodes of degree $\leq n^{\frac{1}{k}}$, and any edges incident to that node, until no such nodes exist. The total number of edges removed in this way is at most $n^{\frac{1}{k}}$, which means that at least $9n^{\frac{1}{k}}$ edges remain (and so the graph is not empty).

The minimum degree of the resulting subgraph is greater than $4n^{\frac{1}{k}}$. If we consider a BFS search from some node $v$ and look at all the levels up until level $k$, if there is no cycle of length $\leq 2k$, then up until level $k$ all edges seen by the BFS form a tree. However, since the branching factor is more than $n^{1/k}$ for each of the levels from 0 to $k - 1$, and so more than $n$ nodes are seen. This is a contradiction. □

The subgraph returned by Create-Spanner is a $(2k - 1)$-spanner by Claim 4, and has $O\left(n^{1 + \frac{1}{k}}\right)$ edges by Claim 5, Claim 6. This completes the proof of the theorem. □

References