Expander Graphs and their Applications*
Lecture 0: How to write the notes

Notes taken by Boaz Barak

March 2002

Summary: This document describes how to write lecture notes for the course.

1 Instructions

Note: This document only explains the technical side to writing the notes using LaTeX. For writing style issues you would have to look elsewhere. If you want to learn by examples you can take a look at the lecture notes for Oded Goldreich's complexity course on `http://www.wisdom.weizmann.ac.il/~oded/cc.html`.

You should write the notes using LaTeX. If you work on a PC you can get LaTeX from `www.miktex.org` and a LaTeX editor from `www.winedt.com`. (For more info about LaTeX look at the web. In particular some links can be found from my home page on `http://www.wisdom.weizmann.ac.il/~boaz`)

To write the notes you should modify the file `lecture.tex`. You should do the following:

1. Change the first three macros in the file (lines 4-6) to the current lecture number, the lecture's title and your name(s). For example in the current document these macros are defined as follows:

```
\def\lectureNr{{0}}   %enter number!
\def\lecture{How to write the notes} %select and enter title
\def\scribe{Boaz Barak}           %enter your name
```

2. write the notes between the comments[1]

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% BEGINNING OF DOCUMENT
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

and

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% END OF DOCUMENT
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

When you download this file, it will contain the current document. You can delete everything between these two comments. Just start your text with a summary (written between `\begin{summary}` and `\end{summary}`).

* Lecture Notes for a course given by Nati Linial and Avi Wigderson at the Hebrew University, Israel.
[1] LaTeX comments start with a % at the beginning of the line.

2 Examples

This section contains some examples of how to use the supplied macros.

The **Theorem** environment allows to write theorems. Use the commands `\begin{theorem}` and `\end{theorem}`. For example:

Theorem 1. *A bipartite graph $G = (V_1, V_2, E)$ has a perfect matching if and only if for any set $S \subseteq V_1$ it is the case that $|\Gamma(S)| \geq |S|$.*

was written using the following commands:

```
\begin{theorem} A bipartite graph $G = (V_1,V_2,E)$ where $|V_1|=|V_2|=n$, has
a perfect matching if and only if for any set $S \subseteq V_1$ it
is the case that $|\Gamma(S)| \geq |S|$.
\end{theorem}
```

There are similar environments for lemmas, claims, definitions, corollaries, conjectures, constructions, examples, and algorithms.

There's also a **proof** environment for the proofs of theorems and lemmas. It is used in the following way:[2]

Proof. For $x \in V_1, y \in V_2$, define $x \prec y$ if $(x, y) \in E$. This is a partial order, where any chain in this order is an edge. By Dilworth's Lemma there is a covering of $V_1 \cup V_2$ by n chains (i.e., a perfect matching) if there is no anti-chain of length $n + 1$. Note that if $A = A_1 \cup A_2$ is an anti-chain (where $A_i = A \cap V_i$) then $\Gamma(A_1) \cap A_2 = \emptyset$. Yet this is impossible for A such that $|A| = n + 1$ since

$$|\Gamma(A_1)| + |A_2| \geq |A_1| + |A_2| > n = |V_2|$$

□

which was produced using

```
\begin{proof} For $x \in V_1,y\in V_2$, define $x \prec y$
if $(x,y) \in E$. This is a partial order, where any chain in this
order is an edge. By Dilworth's Lemma there is a covering of $V_1
\cup V_2$ by $n$ chains (i.e., a perfect matching) if there
is no anti-chain of length $n+1$. Note that if $A = A_1 \cup A_2$
is an anti-chain (where $A_i = A \cap V_i$) then $\Gamma(A_1)
\cap A_2 = \emptyset$. Yet this is impossible for $A$
such that $|A|=n+1$ since
\[
|\Gamma(A_1)| + |A_2| \geq |A_1| + |A_2| > n = |V_2|
\]
\end{proof}
```

There's also a **proof sketch** environment.

Proof Sketch: Use the Min-Cut Max-Flow Theorem. □

which was produced using

```
\begin{proofsk} Use the Min-Cut Max-Flow Theorem.
\end{proofsk}
```

You can see more macros by examining this file. If you want to define your own macros then use your initials as a prefix. For example, I would define the macro

```
\newcommand{\BBLa}{\Lambda}
```

instead of

```
\newcommand{\La}{\Lambda}
```

[2]This is not intended as an example for how to write the proofs themselves!