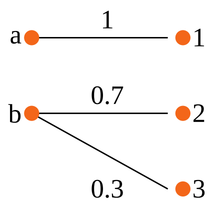
Introduction: Zero-Error Channel Coding

We consider the problem of using a noisy channel to transmit a message perfectly, i.e., with maximal probability of error equal to zero. For some channels, for example the non-trivial binary symmetric channel with $f \notin \{0, 1\}$, it is not possible to send multiple different messages over the channel in this way. For other channels, an interesting question is: how many messages (or how much information) can be sent over this channel in an error-free way?

Example

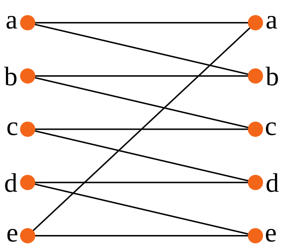
Consider the following channel:



We can send two messages, m_1 and m_2, over the channel by defining $\mathtt{enc}(m_1) = a$ and $\mathtt{enc}(m_2) = b$. The decoding is defined as $\mathtt{dec}(1) = m_1$, and $\mathtt{dec}(2) = \mathtt{dec}(3) = m_2$.

Example: Noisy typewriter

The noisy typewriter channel sends the letters a through e, but with some nonzero probability, it sends the adjacent letter instead. It is defined as follows:



How many messages can you send error-free over this channel?

Show solution

There is a way to send two messages m_1 and m_2 error-free over this channel by defining $\mathtt{enc}(m_1) = a$ and $\mathtt{enc}(m_2) = c$. The decoding function is defined as

$\texttt{dec}(a) = \texttt{dec}(b) = m_1$, and $\texttt{dec}(c) = \texttt{dec}(d) = m_2$ (note that the definition of $\texttt{dec}(e)$ is irrelevant, as this output symbol will never be observed).

Is there a way to encode three different messages in an error-free way? Upon inspection, we see that any encoding function $\texttt{enc}(\cdot)$ on three messages will map at least two messages to channel inputs that are **confusable** (i.e., are possibly mapped to the same channel output).

In general, it is not easy to tell directly from the channel how many messages can be perfectly transmitted. We will invoke some graph theory to help us with the analysis.