به نام خدا

کدینگ منبع

- کدهای با طول ثابت
- کدهای با طول متغیر

از کدهای با طول متغیر می‌توان به کدهای زیر اشاره کرد.

1- کد شانون - فانو (Shannon - Fano)

2- کد هافمن کدی کد منبع بهینه است و به همه آنتروپی منبع می‌رسد. Huffman

طول کلمه

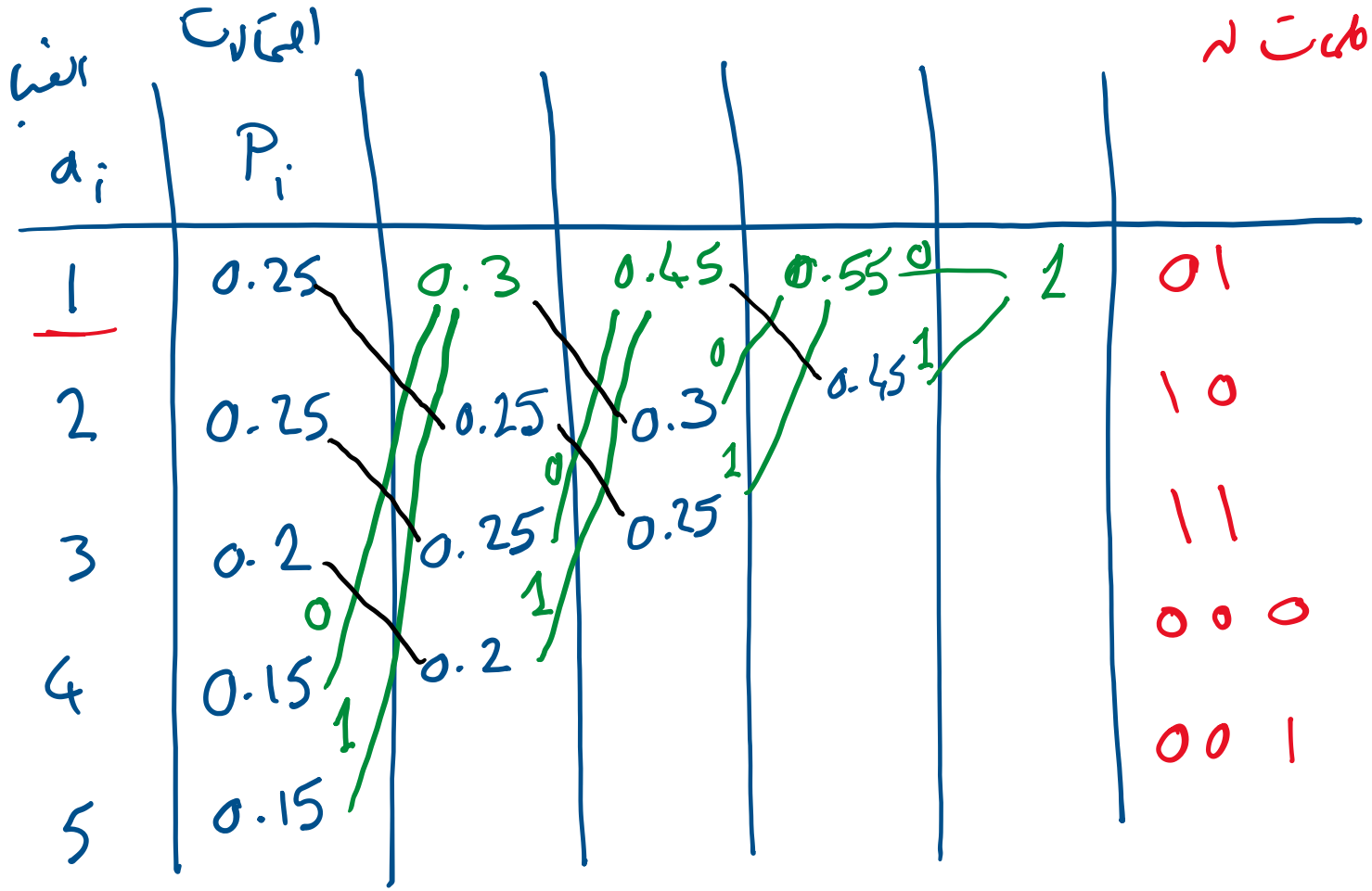
$$d_i = \left[\log \frac{1}{P_i}\right]^+$$

الگوریتم کدینگ هافمن براساس ساختار درختی است که با کمک یک مثال، آن را توضیح می‌دهیم.

مثال: فرض کنیم یک منبع اطلاعات الفبایی از مجموعه $\{1,2,3,4,5\}$ را با احتمالات $\{0.25, 0.25, 0.2, 0.15, 0.15\}$ تولید می‌کند. کدینگ هافمن را برای این منبع به دست آورید. (کدباینری)

۱- ابتدا پیام‌های منبع را به ترتیب نزولی احتمال در ستون‌های ابتدایی جدول کد قرار می‌دهیم.

MSB First



پیامهای با احتمال بیشتر طول کد کمتری دارند

جمیع کلمه‌های کدی باید کلمه‌های کد دیگری نباشد

۲- در هر مرحله دو پیام با احتمال کمتر را با هم ترکیب می‌کنیم و به پیام جدید با احتمال برابر با حاصلجمع پیامهای ترکیب شده می‌سازیم و در مرحله بعد به ترتیب نزولی قرار می‌دهیم. در این مرحله

جا به جایی بیت‌ها را با شاخه‌های درخت مشخص می‌کنیم و بیت‌هایی که با هم ترکیب شده‌اند را با شاخه‌هایی با خروجی‌های '0' و '1' (در حالت باینری) مشخص می‌کنیم.

۳- این روند را ادامه می‌دهیم تا مشخص شود پیام با احتمال یک یا باینری بماند.

۴- برای تعیین کلمه‌ی کد مربوط به هر پیام، از آن پیام شروع می‌کنیم و روی درخت حرکت می‌کنیم و هر جا که شاخه‌ی درخت، سمت راست، آن سمت را به عنوان بیت‌های کلمه‌ی کد، در نظر می‌گیریم. نوشتار کلمه‌ی کد را به صورت MSB first در نظر می‌گیریم.

↓

Most Significant bit / LSB : Least Significant bit

در ارتباط با کدینگ های حاصل به نکات زیر نیز باید توجه داشت:

۱- اگر نخواهیم کلمات کد را M-ary طراحی کنیم، در هر حرکه باید M پیام با کمترین احتمال را با هم ترکیب کنیم. مثلاً اگر بخواهیم کد ternary داشته باشیم، در هر حرکه سه پیام با کمترین احتمال را با هم ترکیب می کنیم و روی شاخه های آنها، ارقام digit ۰، ۱، ۲ قرار می دهیم.

۲- در حالت M-ary اگر در حرکه ای، تعداد پیام ها قبل از پایان الگوریتم، کمتر از M بود، به اندازه ای که به M برسیم، پیام Dummy با احتمال صفر اضافه می کنیم.

تمرین: در کدهای مثال قبل، میانگین طول کلمات کد را محاسبه کنید و با آنتروپی منبع مقایسه کنید.

$$\bar{\ell} \geq H(X)$$

میانگین طول کلمات کد

کُدی رمزی که در مبحث نظری اطلاعات مطرح می‌شود، ضرفت کانال است. این کد را اولین بار شانون با بیان ریاضی، کُد قضیه‌ی معروف شانون، مطرح کرد. براساس قضیه شانون، ماکزیمم نرخ ممکن برای ارسال اطلاعات روی یک کانال مخابراتی را ظرفیت

کانال می کنند و اگر نرخ ارسال کمتر یا مساوی ظرفیت کانال باشد، می توان با خطای به
میزان دلخواه کم، اطلاعات را روی کانال ارسال کرد.

$$\text{نرخ ارسال} \leq \text{ظرفیت کانال}$$

در این قضیه، شانون بیان می کند که راه عملی برای رسیدن به ظرفیت کانال، استفاده از یک
تصویر مناسب از فضای پیام به یک فضای کد است. این تصویرسازی همان کدینگ کانال
است که دنباله های معدی آن را همزمان فراهم کرد.

کانال های ارتباطی را می‌توان به دو دسته‌ی کلی کانال‌های گسسته و کانال‌های پیوسته، تقسیم بندی کرد.

- کانال‌های گسسته، کانال‌هایی هستند که ورودی و خروجی آن‌ها، متغیری از یک مجموعه‌ی گسسته را اختیار می‌کنند.

- کانال های پیوسته، کانال‌هایی هستند که ورودی و خروجی آن‌ها، متغیری از یک مجموعه‌ی پیوسته را اختیار می‌کنند.

* در هر صورت، ظرفیت کانال به سرعت تولید اطلاعات از سمت ورودی کانال قابل تعریف است.

یعنی داریم:

$$C = \max I(X,Y)$$

که در آن X ورودی کانال و Y خروجی کانال است.

$I(X,Y)$ تابع اطلاعات متقابل X, Y است که در درس تئوری اطلاعات به صورت کامل بحث و بررسی می‌شود.

$$I(X,Y) = H(X) - H(X|Y)$$

$H(X|Y)$ → آنتروپی شرطی

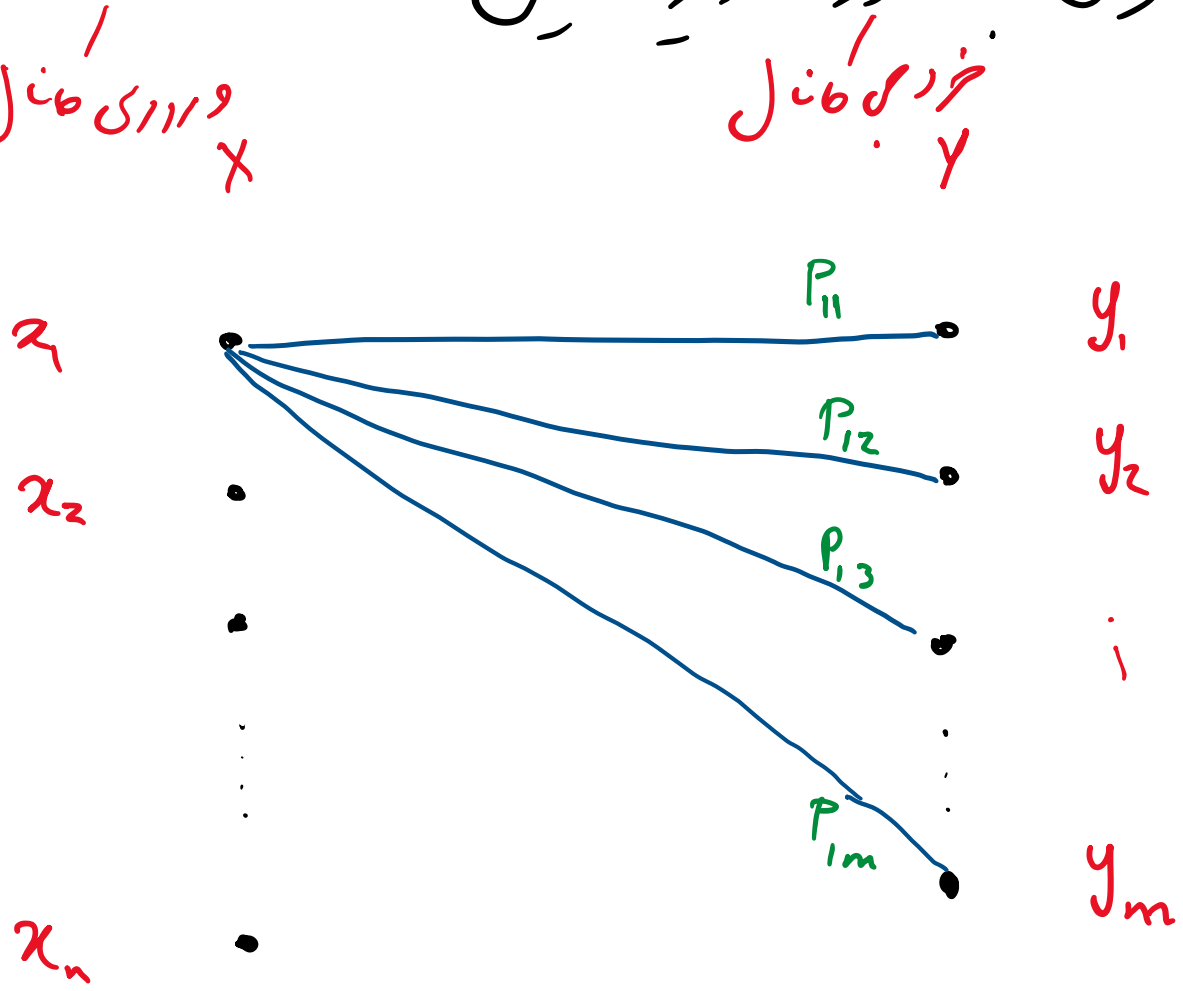
$$= H(Y) - H(Y|X)$$

$$= H(X) + H(Y) - H(X,Y)$$

$H(X,Y)$ → آنتروپی مشترک

مثالهایی از کانال‌های گسسته

* یک کانال گسسته را در کلی‌ترین حالت، می‌توان به صورت زیر نمایش داد



احتمال شرطی $P_{ij} = P_r\{y_j \mid x_i\}$

$= P_r\{$ دریافت y_j | ارسال $x_i\}$

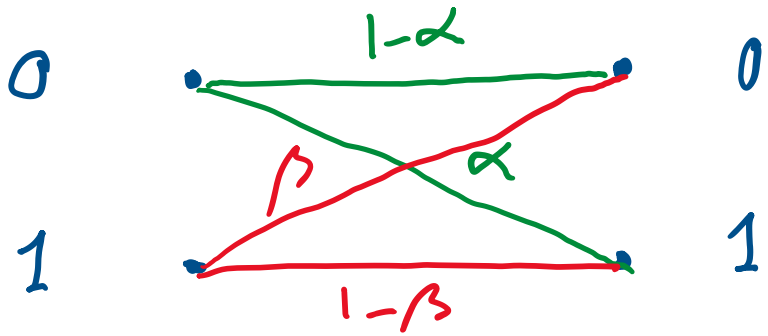
$$\sum_{j=1}^{m} P_{ij} = 1$$

$i = 1, 2, \dots, n$

براین اساس، کانال‌های گسسته با الفبای خروجی، الفبای ورودی و ماتریس گذار، مشخص می‌شوند

ماتریس گذار $P = [P_{ij}]_{n \times m} = \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1m} \\ P_{21} & P_{22} & \cdots & P_{2m} \\ \vdots & & & \\ P_{n1} & P_{n2} & \cdots & P_{nm} \end{bmatrix}$

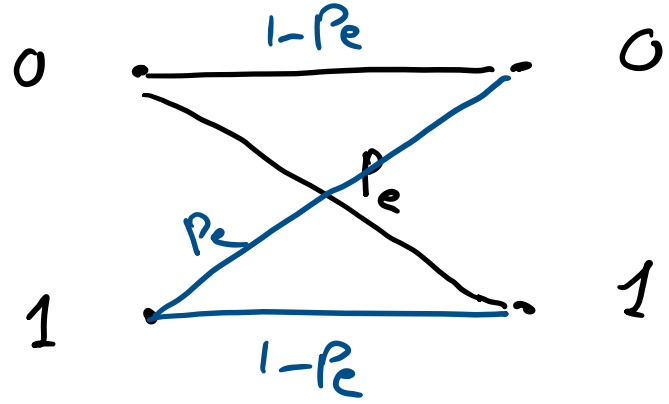
به عنوان مثال: یک کانال باینری را می‌توان به صورت زیر نمایش داد



$P = \begin{bmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{bmatrix}$

* اگر $\alpha = \beta = P_e$ در نظر بگیریم، کانال را یک کانال باینری متقارن گوییم.

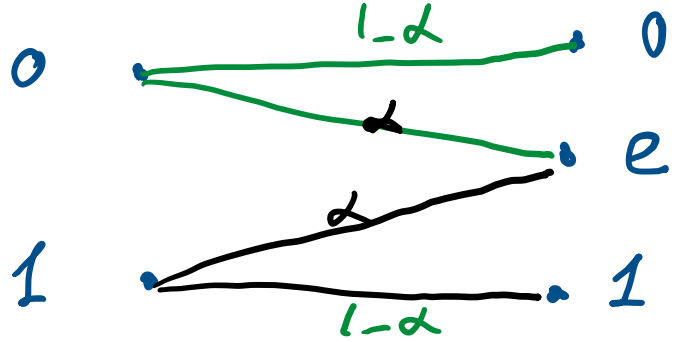
BSC : Binary Symmetric Channel



$$P = \begin{bmatrix} 1-P_e & P_e \\ P_e & 1-P_e \end{bmatrix}$$

جمع سطرها هم برابر یک است.

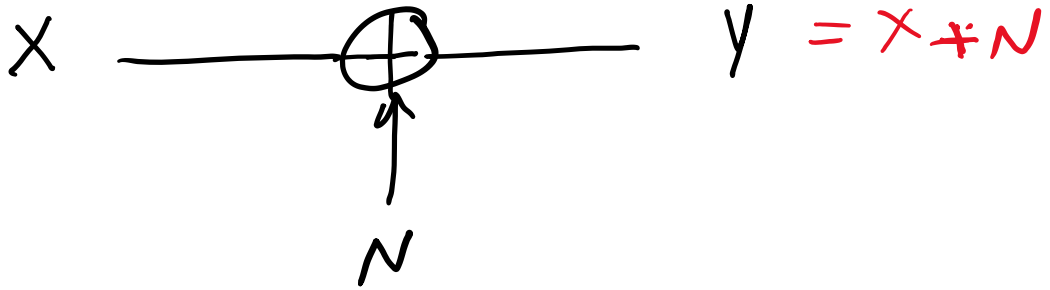
BEC : کانال باینری حذفی یا Binary Erasure Channel



$$P = \begin{matrix} \\ 0 \\ 1 \end{matrix} \begin{bmatrix} 1-\alpha & \alpha & 0 \\ 0 & \alpha & 1-\alpha \end{bmatrix}_{2\times 3} \quad \begin{matrix} 0 & e & 1 \end{matrix}$$

به عنوان یک مثال دیگر به صورت زیر است.

به عنوان مثال حاصل از کانال های پیوسته، می توان به کانال های باند محدود جمع شونده ی پیوسته و وروردی های پیوسته متداول اشاره کرد.



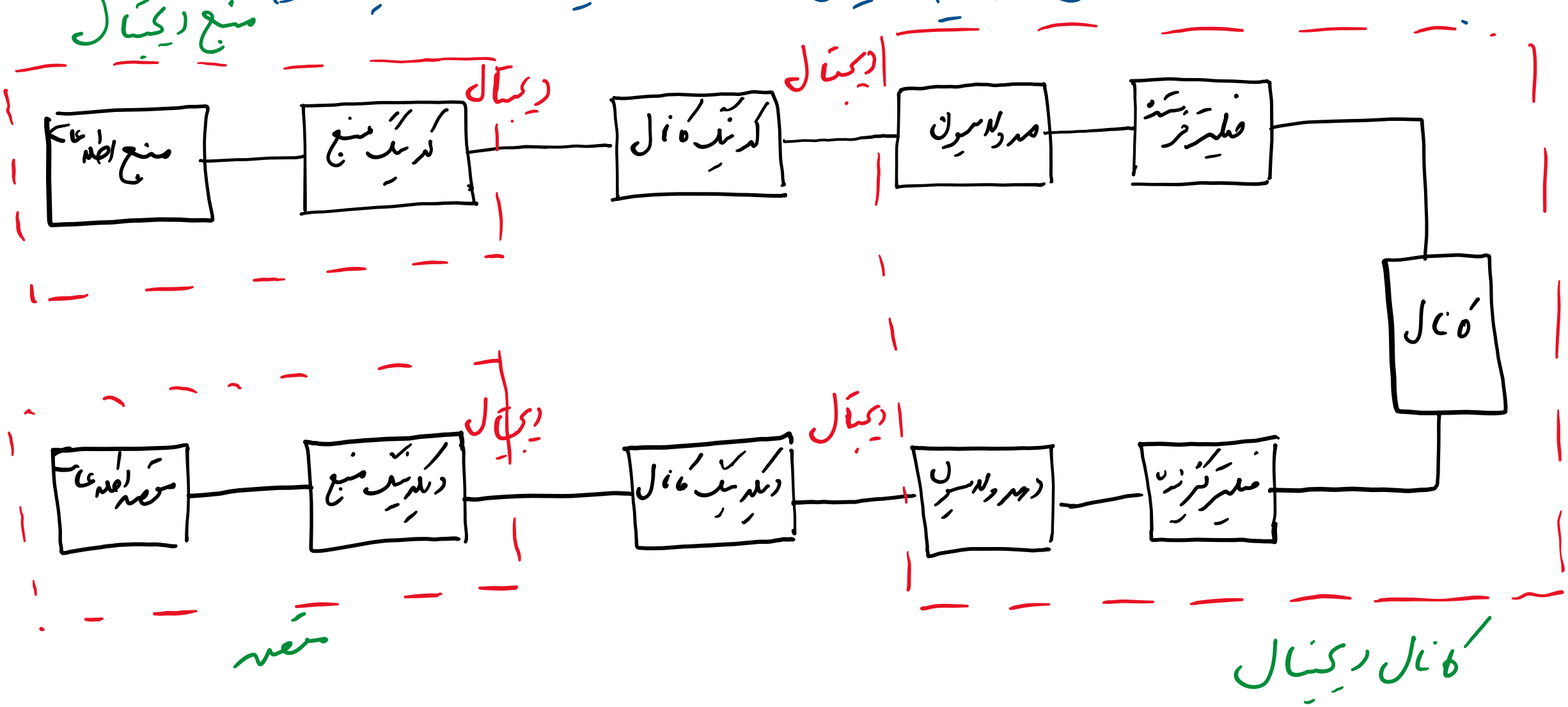
که متداول ترین کانال های جمع شونده در مخابرات، کانال جمع شونده ی گوسی است.

یعنی داریم:

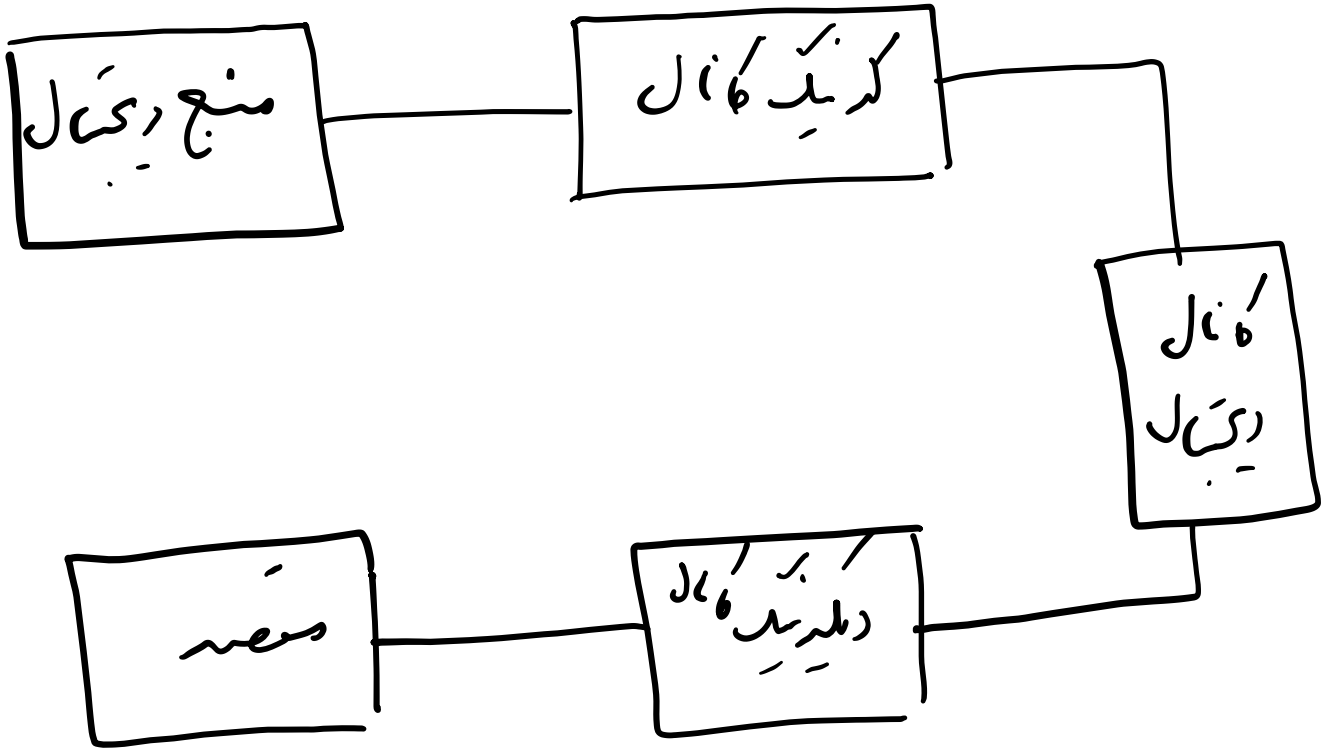
$$N \sim N(m_n, \sigma_n^2)$$

$$C = B \log\left(1 + \frac{S}{N}\right)$$

همان طور که اشاره شد، راه عملی برای رسیدن به طراحی کانال، استفاده از کدینگ کانال مناسب است. در ادامه می‌خواهیم کلیاتی را در مورد کدینگ کانال بیان کنیم.



در محیط گرینک کانال ، بلوک های مدولاسیون کانال در مدولاسیون را به صورت یک کانال دیجیتال مدل سازی می کنیم در ورودی و خروجی انکودر و دیکودر کانال را دیجیتال در نظر می گیریم



در محیط گرینک کانال از بلوک دیاگرام های متفاوت برای تجزیه و تحلیل روش های گرینک کانال استفاده می شود.

همان طور که اشاره شد، انکودر کانال مقداری اضافات کنترل شده، به پیام ارسالی افزایه
که دریافت کننده (دیکودر کانال) از این اضافات برای کاهش یا حذف خطاهای کانال
استفاده می شود. بنابراین مقداری از پهنای باند سیستم به علت این اضافات
Redundancy
مصرف می شود ولی در عوض قابلیت اطمینان سیستم افزایش می یابد.
هدف در کدینگ کانال این است که کدهایی طراحی کنیم که با پیچیدگی قابل قبول
و بهترین استفاده از پهنای باند (بیشترین نرخ ممکن) ما را به قابلیت اطمینان مورد نظر
(احتمال خطای مورد نظر) برسانند. در واقع این پارامترها، پارامترهای اساسی در طراحی

انگیزه و دلیل هستند. احتمال خطا به عنوان معیار ارزیابی کارآیی انگیزه و دلیل مورد توجه قرار می‌گیرد.

برای نشان دادن اهمیت و نحوه‌ی عملکرد کدینگ کانال از یک مثال ساده کمک می‌گیریم.

فرض کنیم که در پیام دلیه و خیر برای فرستادن بردی یک کانال BSC ارسال کنیم. احتمال خطای کانال BSC برابر P_e در نظر می‌گیریم.

* اولین راه حل برای ارسال این پیام‌ها، این است که متناظر پیام دلیه را با '1' و پیام خیر را با '0' که کرده و بیت یا ارتعاره از کانال، ارسال کنیم.

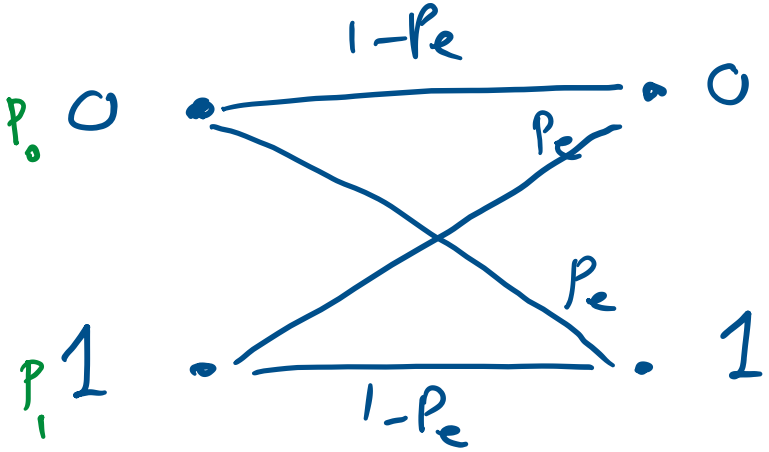
در این حالت:

بله ⟶ 1

خیر ⟶ 0

* نرخ ارسال $R = 1 \ \frac{bit}{channel\ use}$

* احتمال خطا در دریافت اطلاعات P_e



$$P(e) = \underbrace{P_0}_{\text{احتمال خطا در دریافت}} \underbrace{P_r\{1|0\}}_{P_e} + P_1 \underbrace{P_r\{0|1\}}_{P_e} = P_e \overbrace{(P_0+P_1)}^{1} = P_e$$

* آیا گیرنده می‌تواند حدس درستی در مورد وقوع خطا بزند یا در صورت حدس زدن وقوع خطا، آن را تصحیح کند؟ قابلیت تشخیص یا تصحیح خطا دارد؟ خیر

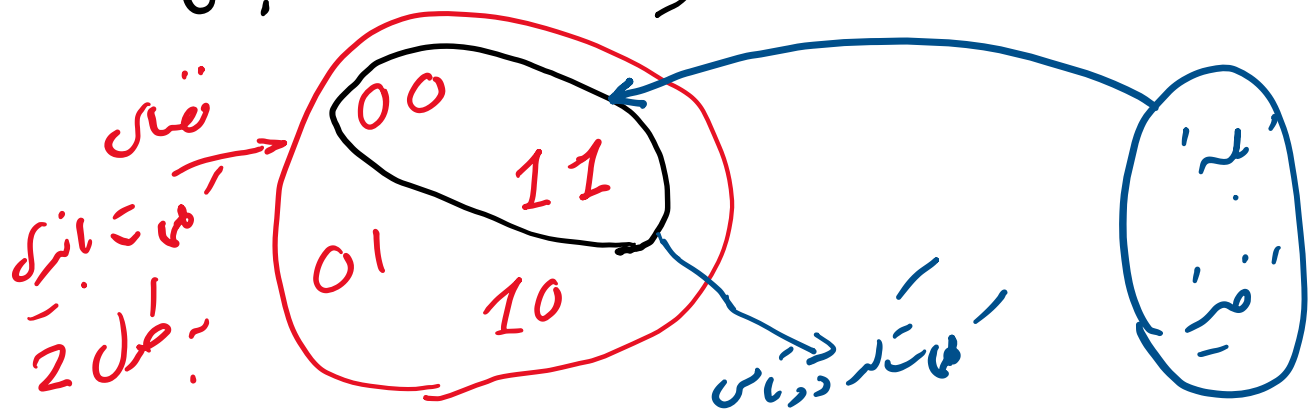
راه حل دوم: پیام "بله" را با "11" و پیام "خیر" را با "00" کد می‌کنیم و بار دیگر از کانال، ارسال می‌کنیم.

"بله" ⟵ 11

"خیر" ⟵ 00

* نرخ ارسال $R = \frac{1}{2}$ $\frac{bit}{channel\ use}$

در این حالت، نسبت به راه حل اول، نرخ ارسال کاهش پیدا کرده است ولی در عوض گیرنده قابلیت تشخیص خطا دارد. زیرا اگر پیام‌های "01" و "10" دریافت کند، متوجه می‌شود که خطایی در کانال رخ داده است.



امکان‌پذیر، قابلیت تصحیح این خطای مشخص داده شده را ندارد، زیرا

- احتمال دریافت '01' به شرط اینکه پیام ۵۵ ارسال شده باشد: $(1-P_e)P_e$
- احتمال دریافت '10' به شرط اینکه پیام ۵۵ ارسال شده باشد: $P_e(1-P_e)$
- احتمال دریافت '00' به شرط اینکه پیام ۱۱ ارسال شده باشد: $P_e(1-P_e)$
- احتمال دریافت '10' به شرط اینکه پیام ۱۱ ارسال شده باشد: $(1-P_e)P_e$

به علت اینکه تمام احتمالات بالا، با هم برابرند، گیرنده نمی‌تواند حدسی در مورد اینکه کدام یک از خطاهای بالا در کانال رخ داده است، تصمیم‌گیری کند.

P_e = احتمال خطای دریافت

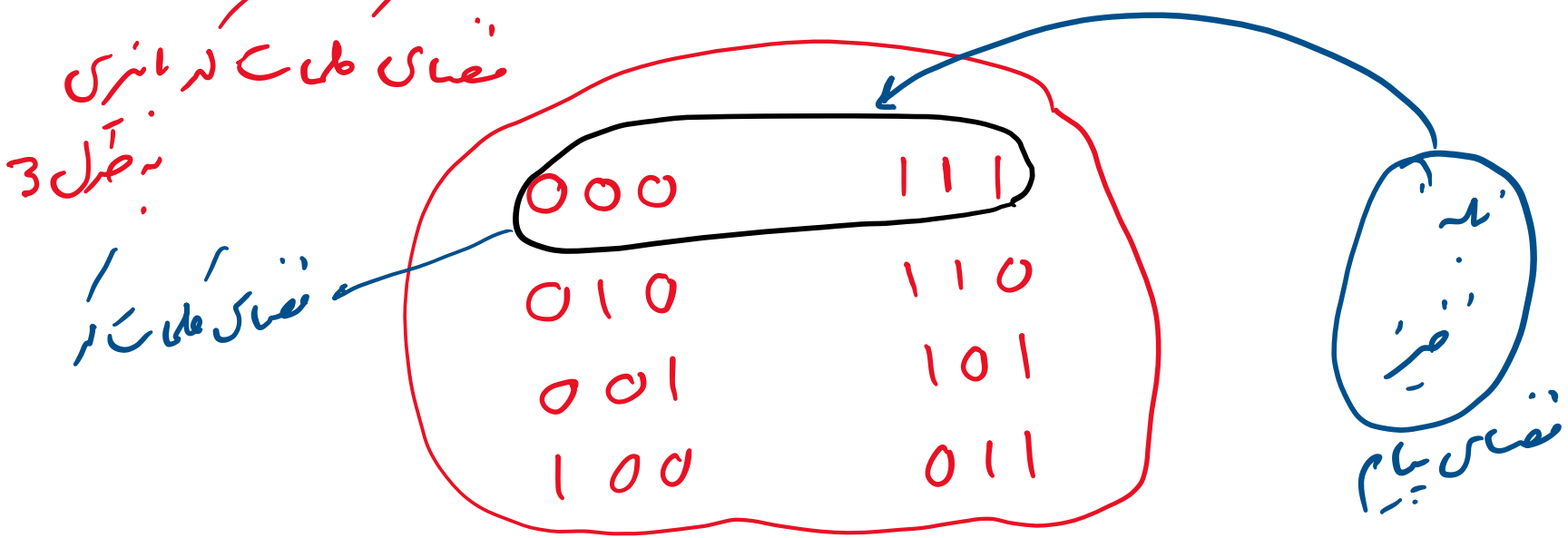
راه حل سوم: پیام "بله" را با "۱۱۱" و پیام "خیر" را با "۰۰۰" کدی می کنیم و با سه بار استفاده از کانال، ارسال می کنیم.

"بله" ⟶ ۱۱۱

"خیر" ⟶ ۰۰۰

* نرخ ارسال $R = \frac{1}{3}$ bit / Channel use

نرخ ارسال سنت به حالت های قبل کاهش یافته است ولی در عوض گیرنده قابلیت تشخیص و تصحیح خطا را نیز دارد.



گیرنده‌ها دریافت یک کلمه‌ی به طول 3 ، با در نظر گرفتن مخلصه‌ترین حالت (که خطای به تعداد یک بیت است) پیام را رسالی را صدق می‌زنند.

- احتمال اینکه یک بیت خطا در کلمه‌ی ارسالی رخ دهد $P_e (1-P_e)^2$

- احتمال اینکه دو بیت خطا در کلمه‌ی ارسالی رخ دهد $P_e^2 (1-P_e)$

- احتمال اینکه سه بیت خطا در کلمه‌ی ارسالی رخ دهد P_e^3

با توجه به اینکه $P_e < \frac{1}{2}$ برقرار است ، مخلصه‌ترین حالت این است که یک بیت خطا رخ دهد.

بنابراین گیرنده براساس محتمل‌ترین خطا در کانال، تصمیم‌گیری می‌کند. به‌این صورت که اگر تعداد '۰' ها در پیام دریافتی بیشتر بود، حدس می‌زند که پیام فرستاده (۰) ارسال شده است، اگر تعداد '۱' ها در پیام دریافتی بیشتر بود، حدس می‌زند که پیام '۱' ارسال شده است.

به‌این ترتیب احتمال خطا در حدس زدن، کمترین مقدار ممکن خواهد بود. (حدس زدن بر اساس بیشترین شباهت)

← برای به دست آوردن قابلیت اطمینان (قابلیت تشخیص و تصحیح خطا)، از نرخ و پیچیدگی خرج کرده‌ایم. هزینه‌ی قابل تبدیل است که با کمترین پیچیدگی و بیشترین نرخ ممکن

ما را به احتمال خطای مورد انتظار در سیستم برسانند.

در مثال قبل اگر طول کلمات کد را به سمت بی نهایت میل دهیم، احتمال خطای را به سمت صفر میل می کند ولی در عوض نرخ ارسال به شدت کاهش می یابد و پیچیدگی سیستم زیاد می شود.

← در ثبت کد یک کانال به طراحی مناسب کد برای رسیدن به قابلیت اطمینان با پیچیدگی قابل قبول در بیشترین نرخ ممکن، پرداخته می شود.