בדיקה הסתברותית של הוכחות | 67790

הרצאות | פרופ׳ גיא קינדלר

כתיבה | נמרוד רק

תשפ״ג סמסטר ב׳

השבוע הראשון הושלם באדיבותו (הרבה) של דויד קיסר-שמידט וסיכומו.

תוכן העניינים

שבוע $\mathbb{I}$ | מבוא

הגדרה <u>מכונת טיורינג</u> היא אוטומט עם סרט זיכרון שהיא יכולה לנוע עליו. מ"ט M מקבלת שפה $L \subseteq \Sigma^*$ אם היא מסיימת במצב מקבל על x אם"ם $x \in L$.

הגדרה <u>מ"ט חישוב</u> זו מ"ט שיש לה מצב עוצר שכשהיא מגיעה אליו הערך שרשום על הסרט הוא הפלט שלה.

הגדרה $\mathsf{P} = \{L :$ קיימת מ"ט המכריעה את L בזמן פולי' $\}$.

הגדרה נאמר כי $L \in \mathsf{NP}$ אם קיימת שפה L^π כך ש:

1. $L^\pi \in \mathsf{P}$.
2. המילים ב-L^π הן מהצורה (x, w) כאשר $x \in L$ ו-$|w| \le \mathsf{poly}(x)$.
3. לכל $x \in L$ קיים w כך ש-$(x, w) \in L^\pi$.

הערה בעיות הכרעה של שפה L הן למעשה חלוקה של Σ^* ל-$(\mathcal{Y}, \mathcal{N})$.

הגדרה <u>בעיית הבטחה</u> (promise problem) היא חלוקה $L = (\mathcal{Y}, \Sigma^* \setminus (\mathcal{Y} \cup \mathcal{N}), \mathcal{N})$ של Σ^*. מ"ט שמזהה את L מקבלת ודוחה נכונה מילים ב-$\mathcal{Y}, \mathcal{N}$ בהתאמה (מבטיחה את התשובה עליהם) ומילים ב-$\Sigma^* \setminus (\mathcal{Y} \cup \mathcal{N})$ יכולות להתקבל, להדחות או שהמ"ט לא תעצור (אין ערובה לתוצאת הריצה).

הערה בעיית הכרעה של שפה L היא בעיית הבטחה מהצורה $(L, \varnothing, \Sigma^* \setminus L)$.

הערה רדוקציה חשיבה לבעיות הבטחה מוגדרת בדומה לרדוקציה בבעיות הכרעה.

הגדרה נאמר כי $L \in \mathsf{NPH}$ אם לכל $L' \in \mathsf{NP}$ קיימת רדוקציה פולי' מ-L' ל-L (כאשר L בעיית הבטחה).

הגדרה נאמר כי $L \in \mathsf{NPC}$ אם $L \in \mathsf{NPH}$ וגם $L \in \mathsf{NP}$.

הגדרה בעיית $\mathsf{MAX} - \mathsf{3SAT}$ מקבלת קלט חוקי $I \in \mathsf{3CNF}$ (נוסחה המורכבת מהסגרים) והמטרה היא לתת השמה שתספק כמה שיותר הסגרים.

עבור קלט חוקי I נגדיר $\mathrm{opt}(I)$ (מסומן לעתים $\mathrm{val}(I)$) האחוז המקסימלי של הסגרים שניתן לספק ב-I.

הערה $\mathsf{3SAT} \in \mathsf{NP}$ היא שפת כל הקלטים החוקיים שהערך שלהם הוא 1 (נוסחה הניתנת לסיפוק במלואה).

הערה $\mathsf{MAX} - \mathsf{3SAT}$ אינה בעיית הכרעה או הבטחה ולכן לעת עתה ההוכחה (העד) אינה מוגדרת היטב.

הערה ל-$\mathsf{3SAT}$ יש כמה מאפיינים מיוחדים מבחינת בדיקת הוכחות. ראשית ניתן לבדוק הוכחה במקביל על כל ההסגרים אם נתון לנו כוח חישוב מקבילי מספיק. ניתן לנצל מנגנון זה לצורכי בדיקה הסת' של השמה: אם $I \in \mathsf{3SAT}$ אז בהגרלת הסגרת, ההסת' שיסופק היא $P(\mathcal{Y}) = 1$ אבל אם $I \notin \mathsf{3SAT}$ אז $P(\mathcal{Y}) \le 1 - \frac{1}{m}$ (לפחות הסגר אחד לא מסופק). כלומר ניתן להגדיר מוודא הסת' לבעיה.

הגדרה מוודא הסת' לבעיית הבטחה הוא מ"ט שמקיים את התנאים הבאים:

- (לוקליות) המ"ט מבצעת מספר גישות קבוע לעד (3 ביטים בלבד מתוך העד).
- (רנדומיות) המ"ט מגרילה $\mathcal{O}(\log n)$ ביטים.
- (שלמות) המ"ט מקבלת קלט בשפה בהסת' 1 (המוודא מושלם).
- (תקפות, Soundness) קיים לו חסם מלעל להסת' לקבלת קלט שאינו בשפה (במקרה שלנו $1-\Theta\left(\frac{1}{n}\right)$).

טענה לכל $L \in \mathsf{NP}$ קיים מוודא הסת' עם פרמטרים כמו שכתבנו למעלה.

הוכחה: ממשפט קוק-לוין, יש רדוקציה מ-L ל-3SAT ולכן מספיק לבדוק הסת' את הקלט המתקבל ל-3SAT. ■

משפט (PCP בניסוח 3SAT) לכל $L \in \mathsf{NP}$ קיים מוודא הסת' עם פרמטרים כנ"ל ו-$1 >$ soundness $<$ const (ישנו חסם מלעל קבוע קטן ממש מאחד לתקפות).

הערה כדי לקיים את הדרישה על התקפות צריך שהרדוקציה מהשפה לנוסחה ב-3CNF תיתן נוסחה שהיא בהסת' נמוכה ספיקה.

הגדרה $\mathsf{gap}-\mathsf{MAX}-\mathsf{3SAT}\,[c,s]$ היא בעיית ההבטחה עם

$$\mathcal{Y} = \{I : \text{קלט חוקי לבעיית 3SAT } I \ \wedge\ \mathrm{val}(I) \geq c\}$$

$$\mathcal{N} = \{I : \text{קלט חוקי לבעיית 3SAT } I \ \wedge\ \mathrm{val}(I) \leq s\}$$

הערה אינטואטיבית, c הוא המשלים (אחד פחות-) אחוז ה-false negative שאנחנו מוכנים לסבול ו-s הוא אחוז ה-false postiive שאנחנו מוכנים לסבול.

משפט (ניסוח מחדש של PCP עם $\mathsf{gap}-\mathsf{MAX}-\mathsf{3SAT}$) קיים $s<1$ כך ש-$\mathsf{gap}-\mathsf{MAX}-\mathsf{3SAT}\,[1,s] \in \mathsf{NPH}$.

הערה הניסוח החדש מספיק כי ל-$\mathsf{gap}-\mathsf{MAX}-\mathsf{3SAT}\,[1,s]$ יש מוודא הסת' שעונה על הקריטריונים האמורים לעיל ולכן עם רדוקציה מכל $L \in \mathsf{NP}$ נקבל את משפט ה-PCP המקורי.

המוודא מקבל $I = c_1 \wedge \ldots \wedge c_m$ נוסחה חוקית ו-f השמה (העד), מגריל $i \in [m]$ ובודק האם c_i מסופקת ע"י f (צריך לבדוק את שלושת הביטים ב-f המתאימים לליטרלים ב-c_i). אם הפסוקית מסופקת יענה $\mathcal{Y}$ ואחרת $\mathcal{N}$.

- אם $I \in \mathcal{Y}$ אז יש השמה מספקת ולכן המוודא יענה $\mathcal{Y}$ על איזשהו עד (לכן תמיד נסווג נכון $I \in \mathcal{Y}$).
- אם $I \in \mathcal{N}$ אז $s \cdot m$ פסוקיות לכל היותר מסופקות ע"י כל השמה ולכן ההסת' שניפול על אחת מסופקת (שתגרום לנו לחשוב ש-I כן ספיקה) היא s, כלומר s הוא קבוע התקפות במקרה הזה.

הגדרה אלג' α-מקרב ל-$\mathsf{MAX}-\mathsf{3SAT}$ (עבור $\alpha \in [0,1]$) הוא אלג' שמקבל כקלט נוסחת 3CNF חוקית I ומחזיר מספר b שמקיים $\alpha\,\mathrm{val}(I) \leq b \leq \mathrm{val}(I)$.

מסקנה (ממשפט ה-PCP) אם $\mathsf{P} \neq \mathsf{NP}$ אז לא קיים אלג' α-מקרב פולינומי ל-$\mathsf{MAX}-\mathsf{3SAT}$ עבור $\alpha > s$ (כאשר s הקבוע ממשפט ה-PCP).

הוכחה: נניח בשלילה שקיים אלג' כזה. תהי $L \in \mathsf{NP}$, לכן קיימת רדוקציה f מ-L ל-$\mathsf{gap}-\mathsf{MAX}-\mathsf{3SAT}\,[1,s]$. יהי קלט w לבעיית ההכרעה L. נריץ את אלג' הקירוב על $f\left(w\right)$ ונקבל

$$\alpha \; \mathrm{val}\left(f\left(w\right)\right) \le b \le \mathrm{val}\left(f\left(w\right)\right)$$

- אם $w \in L$ אז $b \ge \alpha \underbrace{\mathrm{val}\left(f\left(w\right)\right)}_{\ge 1} \ge \alpha > s$.
- אם $w \notin L$ אז $b \le \mathrm{val}\left(f\left(w\right)\right) \le s$.

כלומר השוואה של b ל-s תכריע האם $w \in L$ ולכן מ"ט דטר' פולי' בזמן יכולה להכריע את L כלומר $L \in \mathsf{P}$, ולכן $\mathsf{P} = \mathsf{NP}$ סתירה. ■

מסקנה אם $\mathsf{gap}-\mathsf{MAX}-\mathsf{3SAT}\,[c,s] \in \mathsf{NPH}$ וגם $\mathsf{P} \neq \mathsf{NP}$ אז אין אלג' קירוב עם פרמטר גדול מ-$\frac{s}{c}$.

הוכחה: כנ"ל. ■

דוגמאות לאלג' קירוב לבעיות קשות ב-NP

- ראינו באלגו' אלג' $\frac{7}{8}$-מקרב ל-3SAT (מגרילים הרבה השמות עד שאחת מספקת לפחות $\frac{7}{8}$ מהפסוקיות).
- בעיית $\mathsf{MAX}-\mathsf{Exact3}-\mathsf{LIN2}$ היא בעיית האופטימיזציה מעל מערכת n משוואות, בכל אחת שלושה משתנים (שניתן לשים בהם $0,1$) שערכה הוא המספר המקס' של משוואות שניתן לספק במערכת.

  אלג' $\frac{1}{2}$-מקרב לבעיה (שראינו באלגו') בודק לכל משתנה איזו השמה עדיפה (לפי תוחלת סיפוק המשוואה) ובוחר באופן חמדני את ההשמה העדיפה.

  ידוע כי $\mathsf{gap}-\mathsf{MAX}-\mathsf{E3}-\mathsf{LIN2}\left[1-\epsilon, \frac{1}{2}+\epsilon\right]$ היא בעיה קשה ב-NP לכל $\epsilon > 0$ (כלומר אינטואיטיבית ממש קשה להבדיל בין מערכות משוואות שניתן לספק כמעט את כל המשוואות בהן לבין מערכות שניתן לספק קצת יותר מחצי ממשוואותיהן).
- בעיית $\mathsf{MAX}-\mathsf{IS}$ לכל גרף מחזירה את גודל קבוצת הקודקודים הבת"ל (אף שני קודקודים בקבוצה אינם מחוברים בצלע) המקסימלית.

  ידוע כי $\mathsf{gap}-\mathsf{MAX}-\mathsf{IS}\left[1-\frac{1}{\sqrt{2}}-\epsilon, \epsilon\right]$ קשה ב-NP לכל $\epsilon > 0$ (ראו הסבר אינטואיטיבי לעיל).

קווים לדמותו של PCP - משחק עם שני שחקנים חזקים ומוודא חלש

נתונים שני שחקנים (חזקים חישובית) שמשחקים משחק: בהינתן נוסחה, הם מתאמים עמדות (בוחרים השמה) ואז מופרדים.

שחקן אחד מקבל פסוקית ושחקן נוסף משתנה בפסוקית. הראשון מחזיר השמה למשתנים בפסוקית והאחרון השמה למשתנה.

הם מנצחים אם ההשמה של הראשון מספקת את הפסוקית ואם שני השחקנים מסכימים על הערך המושב במשתנה שניתן לאחרון מתוך הפסוקית.

הערה הרעיון מאחורי המשחק הזה הוא שקילות ה-PCP למצב בו שני שחקנים חזקים חישובית מנסים להראות הסת' למוודא חלש מאוד שניתן לספק את נוסחה מסוימת.

טענה בהינתן $\mathrm{val}\left(I\right)\leq\alpha$ (שיעור הפסוקיות שניתנות לסיפוק בו זמנית המקסימלי), ההסת' שינצחו היא $P\left(\mathrm{success}\right)\leq1-\frac{1-\alpha}{3}$.

הוכחה: נניח שהשחקנים משחקים באסטרטגיה עם שיעור הצלחה β. לכן

$$\begin{aligned}E_{c\in I}\left[\mathbb{1}_{\{\text{מפסידים על } c\}}\right] & \overset{(*)}{\leq}E_{c\in I}\left[\mathbb{1}_{s_{1}(c)\neq s_{2}(c)}\right]\\
 & \overset{(**)}{\leq}3\cdot E_{c\in I}\left[\frac{\sum_{i=1}^{3}\mathbb{1}_{s_{1}(c_{i})\neq s_{2}(c_{i})}}{3}\right]\\
 & \overset{(***)}{=}3\cdot\left(1-\beta\right)\end{aligned}$$

$(*)$ מונוטוניות ההסת': השחקנים אידאליים ולכן אם הפסוקית ניתנת להשמה תחת ההשמה (אסטרטגיה) שהוסכמה בהתחלה, שניהם ייתנו אותה. אם היא לא מסופקת תחת ההשמה שחקן 1 ידע את זה וישנה את ההשמה (שתספק ובתקווה תהיה זהה להשמת שחקן 2 למשתנה). לכן אם הם מפסידים הם בהכרח לא מסכימים על ההשמה לפסוקית (של שחקן 1 זו החדשה שהמציא עכשיו ממנה הוא חושף 3 ערכים למוודא ושל 2 היא המוסכמת במקור ממנה הוא חושף ערך אחד למוודא). $s_{1}\left(c\right),s_{2}\left(c\right)$ הן וקטורים ב-$\{0,1\}^{3}$.

$(**)$ הכפלה וחלוקה ב-3 וגם חסם האיחוד על אי ההסכמה על ההסגר (לפחות אחד מהליטרלים לא מוסכם).

$(***)$ הצלחה היא לשכנע את המוודא שניתן לספק את הפסוקית (במרמה או לאו), ואי הסכמה יש רק כשההשמה המקורית לא מספקת את הפסוקית (כלומר הנוסחה לא ספיקה). לכן ההסת' לכישלון $1-\beta$ היא ההסת' לאי הסכמה בין השחקנים, שזה בדיוק תוחלת ממוצע אי ההסכמה במשוואה למעלה.

ולכן

$$P\left(\mathrm{success}\right)=\beta\leq1-\frac{E_{c\in I}\left[\mathbb{1}_{\{\text{לא מסופק } c\}}\right]}{3}\leq1-\frac{1-\alpha}{3}$$

■

הגדרה משחק בין שני שחקנים עם סיבוב אחד (2 Player 1 Round Game) הוא שלשה $G=\left\langle V,P_{1},P_{2}\right\rangle$ כאשר:

- $P_{1}=\left\langle Q_{1},\Sigma_{1}\right\rangle ,P_{2}=\left\langle Q_{2},\Sigma_{2}\right\rangle$ הם השחקנים כאשר Q_{1},Q_{2} אוסף שאלות ו-Σ_{1},Σ_{2} אוסף תשובות.
- $V=\left\langle D,P\right\rangle$ הוא מוודא כאשר D התפלגות מעל $Q_{1}\times Q_{2}$ (לא בהכרח ב"ת) ו-P "פרדיקט" שהוא פ' $Q_{1}\times Q_{2}\times\Sigma_{1}\times\Sigma_{2}\mapsto\{0,1\}$.

ערך ההצלחה של המשחק הוא $\mathrm{val}\left(G\right)=\sup_{\text{strategies}}P\left(\mathrm{success}\right)$.

טענה נניח שאנחנו משחקים את המשחק למעלה עם שני השחקנים והנוסחה I שעבורה מתקיים $\mathrm{val}\left(I\right)=\alpha$. אז ניתן לחשב את $\mathrm{val}\left(G\right)$ בזמן סופי.

הוכחה: תוחלת ההצלחה במשחק היא α (ההסת' שניפול על פסוקית שסופקה ע"י ההשמה המקסימלית שלנו) כלומר

$$\alpha = E\left[\mathbb{1}_{\text{success}}\right] = E_{r_1,r_2}\left[E_{\textsf{strategies}}\left[\mathbb{1}_{\text{success}}\right]\right]$$

כאשר r_1, r_2 סרטי ביטים אקראיים (ככה ממודלת גישה לערכים אקראיים), והאסטרטגיות בתוחלת הפנימית למעשה עוברות דטרמיניזציה כי בהינתן סרט עם הערכים האקראיים שלו, האסטרטגיה נהפכת לדטר'. מתכונות התוחלת, יש לפחות אסטרטגיה אחת (א"ד שנהיית דטר' תחת סרט מקרי כלשהו) עם לפחות ערך α, שזה הכי הרבה שאנחנו יכולים להשיג. לכן מספיק שנעבור על כל האסטרטגיות הדטר' ונקבל $\text{val}\left(G\right) = \max_{\text{det}'\ \text{strategies}} P\left(\text{success}\right)$ כלומר שהאסטרטגיה שמשיגה sup היא מתוך קבוצה סופית. ■

שבוע III | קודים לתיקון שגיאות

כל טענה מתמטית ניתן לקודד באופן שמחשב יוכל להבין אותו (מעל א"ב כלשהו), ולכן בהינתן טענה S, נוכל לכתוב הוכחה π שגם אותה נוכל לקודד. מעבר לכך ישנו אלג' שרץ בזמן פולי' (באורך הטענה וההוכחה) שמוודא את ההוכחה. עם זאת מציאת הוכחה לטענה נתונה היא לא כריעה.

טענה בהינתן טענה S וסטרינג אונרי 1^n, הבעיה האם יש תווים שמחליפים את 1^n כך שהם מהווים הוכחה חוקית ל-S, היא ב-NP (אפשר לוודא עד פולי'), ובפרט היא שלמה ב-NP.

מסקנה ממשפט ה-PCP, נוכל לבנות מוודא הסת' שדוגם מספר קבוע של ביטים מהוכחה לטענה מתמטית כלשהי (לא רק נוסחת 3SAT) וקובע האם היא תקינה או לא. כלומר הבדיקה הלוקאלית היא להוכחות כלליות ולא לבעיה ספציפית!

הערה קידוד הוא מחרוזת מוארכת מהמקורית שכולל יתירות כדי שיהיה אפשר לשחזר אותו לאחר שהושחת. קודים הם אוסף הקידודים של המילים (לאחר שקודדו), שמהם אפשר לבחור אחד שיעזור לשחזר תוכן מקורי וכו'.

הגדרה יהי Σ אלפבית. קוד מעל Σ הוא $C \subseteq \Sigma^n$ ויש לו ארבעה פרמטרים (n, d, R, q):

- n - אורך המילים המקודדות (block length).
- d - המרחק של הקוד, שערכו $\min_{u \neq w \in C}\left\{h\left(u, w\right)\right\}$ כאשר $h\left(u, w\right) = P_{i \in [n]}\left(u_i \neq w_i\right)$ (שיעור הקוורדינטות עליהן הוקטורים מסכימים).
- R - הקצב (rate) שערכו $\frac{\log|C|}{\log|\Sigma^n|}$.
- q - גודל הא"ב, $q = |\Sigma|$.

הערה אם u ו-w בקוד מאוד רחוקות אחת מהשנייה לפי מרחק האמינג. אם נשדר את u וחלק מהמידע מושחת כך שהתקבל u', נוכל לשחזר אותה ל-u כי כל מילה אחרת בקוד יותר רחוק מ-u' מאשר u. למעשה כל מרחק פחות מ-$\frac{d}{2}$ ניתן לשחזר נכונה.

הערה הערך העליון בקצב, אם נסתכל בבסיס $|\Sigma|$ נותן לנו את מספר האותיות ב-Σ שנדרשות כדי לייצג את כל המילים ב-C (אם $\log_{|\Sigma|}|C| = 17$ אז נוכל לקודד את כל המילים באורך 17). לכן היחס למעשה מגדיר את היעילות של הקוד - כמה גדול הניפוח ממספר הביטים של האותיות שאנחנו רוצים לייצג ($\log_{|\Sigma|}|C|$) לאורך הקוד שלנו בסוף ($\log_{|\Sigma|}|\Sigma|^n = n$). לכן, R גבוה הוא תכונה רצויה.

הגדרה בהינתן קוד C, $B_{w}^{n}\left(\alpha\right)=\left\{ u\in\Sigma^{n}:h\left(u,w\right)\leq\alpha\right\}$ הוא אוסף המילים במרחק (האמינג) לכל היותר α.

הגדרה עבור $\Sigma=\mathbb{F}_{q}$ (שדה מודולו מעל q ראשוני), $C\subseteq\Sigma^{n}=\mathbb{F}_{q}^{n}$ הוא קוד לינארי אם C הוא מרחב וקטורי (ת"מ של Σ^{n}).

הערה במקרה כזה,

$$d\left(C\right)=\min_{u\neq w\in C}\left\{ h\left(u,w\right)\right\} \overset{(*)}{=}\min_{u\in C\backslash\left\{ 0\right\} }\left\{ h\left(u,0\right)\right\} \overset{(**)}{=}\min_{u\in C\backslash0}\left|u\right|$$

$(*)$ $h\left(u,w\right)=h\left(u-w,0\right)$. $(**)$ כך נגדיר ערך מוחלט.

בנוסף, $R=\frac{\dim C}{n}$ כי כל איבר של C ניתן לייצג ע"י $\dim C$ מספרים (שהם הקוורדינטות של וקטורי בסיס של C).

הגדרה בהינתן בסיס $\left\{ M_{1},\ldots,M_{Rn}\right\}$ (של וקטורים עומדים) לקוד C, $M=\left(M_{1}\ \ldots\ M_{Rn}\right)$ נקראת המטריצה היוצרת של C.

הערה באמצעות המטריצה היוצרת ניתן לקודד בקלות וביעילות ע"י Mx מפני שתמונת M היא C.

טענה $1\geq R+\frac{d}{2}+o_{\left|\Sigma\right|}\left(1\right)$.

הוכחה:

$$\left|\Sigma\right|^{n}\overset{(i)}{\geq}\left|C\right|\cdot\left|B_{0}\left(\frac{d}{2}\right)\right|\overset{(ii)}{\geq}\left|C\right|\binom{n}{\frac{1}{2}dn-1}\left|\Sigma\right|^{\frac{dn}{2}}\overset{(iii)}{\geq}\left|\Sigma\right|^{Rn+\frac{dn}{2}}2^{\mathcal{O}(n)}\overset{\left|\Sigma\right|\to\infty}{=}\left|\Sigma\right|^{Rn+\frac{dn}{2}+o(n)}$$

(i) כל מילה ב-w נמצאת בכדור ברדיוס $\frac{d}{2}$ שבו היא נמצאת ללא מילים אחרות בקוד. לכן נוכל למלא את $\left|\Sigma\right|^{n}$ בכדורים ברדיוס $\frac{d}{2}$ סביב כל המילים ב-C ועדיין לא למלא את כל Σ^{n} (או בדיוק כן למלא).

(ii) המילים ב-$B_{0}\left(\frac{d}{2}\right)$ הם המילים על לכל היותר $\frac{d}{2}$ אותיות שאינם 0. לכן קומבינטורית, נבחר את $\frac{1}{2}dn$ האותיות שנשנה (אחד פחות כדי למנוע התנגשויות), ונקבע בהם את הערכים החדשים (בפרט יכולים להיות גם 0).

(iii) $\log\left|C\right|=Rn$ וחסם עליון ל-choose מהצורה $\binom{n}{\frac{1}{2}dn}$ הוא $2^{\mathcal{O}(n)}$.

ומשם ניקח $\log_{\left|\Sigma\right|}$ על שני האגפים, נחלק ב-n ונקבל את הנדרש. ■

קודי Reed-Solomon

בהינתן שתי פרובולות, אנחנו יודעים שהן נפגשות לכל היותר בשתי נקודות, ולכן מבחינת הערכים שלהן הן די שונות. באותו האופן פולינומים ממעלה נמוכה גם כן כשאינם זהים אינם חולקים ערכים רבים.

הגדרה נקבע את דרגת הפולינומים מעל $\mathbb{F}_{q}$ (q ראשוני כי זה שדה) איתם נעבוד להיות $d\leq n\leq q$ ונבחר $a_{1},\ldots,a_{n}\in\mathbb{F}_{q}$. הקוד של ריד-סולומון הוא

$$RS_{d,a_{1},\ldots,a_{n},q}=\left\{ f\left(a_{1}\right),\ldots,f\left(a_{n}\right)\mid\deg f\leq d\text{ פולינום עם }f:\mathbb{F}_{q}\to\mathbb{F}_{q}\right\}$$

הערה זהו קוד לינארי מסגירות הפולינומים מדרגה לכל היותר d לחיבור.

נחשב את הפרמטרים של הקוד.

- אורך הקוד הוא n.
- מרחק הקוד הוא $1-\frac{d}{n}$ כי שני פולינומים שונים הם לכל היותר שווים ב-d נקודות.
- קצב הקוד הוא $\frac{d+1}{n}$ כי $\dim C=d+1$ וזהו קוד לינארי.
- גודל הא"ב הוא $q=\left|\Sigma_q\right|$.

אם נבחר $d\leq\frac{n}{2}$ נקבל קצב ומרחק $\frac{1}{2}$ שזה מה שרצינו, וגודל א"ב בין n ל-$2n$ (שם בהכרח יש ראשוני).

כרגע יש לנו n^n מילים ב-$|\Sigma|^n$. נרצה משהו עם משמעותית פחות אותיות. אם נבחר קוד עם n מילים, נוכל לבחור כל מילה לייצג אות אחרת ב-Σ וכך לייצג קודים ב-C באמצעות מילים מהקוד הקטן יותר, ובתקווה עדיין לשמר את אותן התכונות.

הרכבת קודים

הגדרה יהיו C_1 קוד $\left(n_1,d_1,r_1,q_1\right)$ מעל Σ ו-C_2 קוד $\left(n_2,d_2,r_2,q_2\right)$ מעל Σ' כאשר $q_2\gg q_1$. נדרוש $\left|C_2\right|\geq q_1$ וקיום $E:\Sigma\overset{\text{חח"ע}}{\to}C_2$ (קידוד אותיות למילים בקוד C_2). נגדיר את <u>ההרכבה של הקודים</u> C_1,C_2 להיות

$$C_1\circ C_2=\left\{\left(E\left(x_1\right)||\ldots||E\left(x_{n_1}\right)\right):x_1\ldots x_{n_1}\in C_1\right\}$$

פרמטרים של הרכבה

- אורך הקוד הוא $n_1\cdot n_2$ (יש לנו n_1 מילים משורשות, כל אחת באורך n_2).
- מרחק הקוד הוא $d\left(C_1\circ C_2\right)\geq d_1\cdot d_2$ כי כשנדגום קוורדינטה מקרית נדגום קודם קוורדינטה מהמילים המקוריות ב-C_1 לפני שתורגמו, שם הסיכוי לשוויון הוא d_1, ואז לאחר שנתרגם הסיכוי לשוויון בקוורדינטה הוא d_2.
- קצב הקוד הוא $R_1\cdot R_2$ (עד כדי קבוע קטן) מהחישוב

$$\begin{aligned}R\left(C_1\circ C_2\right)&=\frac{\log\left|C_1\right|}{\log\left(q_2^{n_1\cdot n_2}\right)}\\&=\frac{\log\left|C_1\right|}{\log\left(q_1^{n1}\right)}\cdot\frac{\log\left(q_1^{\cancel{n1}}\right)}{\log\left(q_2^{\cancel{n_1}\cdot n_2}\right)}\\&=R_1\cdot R_2\end{aligned}$$

כאשר באופן אופטימלי $|C_2|$ קרוב כמה שיותר (מלמעלה) ל-q_1.

הערה הרכבת קידודים לינאריים עם E לינארית היא קוד לינארי.

השגת קוד עם פרמטרים קבועים וא"ב בגודל 2

משפט קיים קוד מעל הא"ב $\{0,1\}$ עם מרחק וקצב קבוע ואורך מילה $\log\log\log n$.

הוכחה: נבחר C_1 קוד עם פרמטרים $\left(n, \frac{1}{2}, \frac{1}{2}, n\right)$ (ריד-סולומון עם $q = n$, $d = \frac{n}{2}$). יש ב-C_1 $n^{\frac{n}{2}}$ איברים, כי כל $\frac{n}{2}$-יה של ערכים ב-$\mathbb{F}_q$ ניתנת להשגה ע"י פולינום ממשפט האינטרפולציה, נניח כי $q = n$.

נבחר C_2 עם פרמטרים $\left(\log n, \frac{1}{2}, \frac{1}{2}, \log n\right)$ (ריד סולומון עם $q = k$ ו-$d = \frac{k}{2}$ עבור $k = \log n$) לכן יש בו $k^{\frac{k}{2}} = |C_2| \geq n$.

עתה $C = C_1 \circ C_2$ הוא קוד עם פרמטרים $\left(n\log n, \frac{1}{4}, \frac{1}{4}, \log n\right)$.

נוכל להפעיל זאת שוב עם C_3 שלו פרמטרים $\left(\log\log n, \frac{1}{2}, \frac{1}{2}, \log\log n\right)$ ונקבל $C \circ C_3$ עם פרמטרים $\left(n\log n\log\log n, \frac{1}{8}, \frac{1}{8}, \log\log n\right)$.

בטווח הארוך אמנם, אנחנו מאבדים ביצועים ולא מתקרבים ל-$q = 2$. נצטרך גישה אחרת.

אם קיים קוד C_4 עם פרמטרים $\left(\log\log\log n, \frac{1}{100}, \frac{1}{100}, 2\right)$ ואז נוכל להרכיב אותו עם $C \circ C_3$ ולקבל קוד עם קצב, מרחק וגודל א"ב קבוע, ומספר מילים בקוד קרוב מאוד ל-n, שזו המטרה הסופית שלנו.

בנוסף, מספר תתי הקבוצות של מילים באורך $\log\log\log n$ מתוך $\{0,1\}^*$ הוא $\log n = 2^{2^{\log\log\log n}}$ כלומר נוכל בזמן פולי' לעשות ברוט פורס על כל הקבוצות עד שנגיע לאחת שהיא קוד עם פרמטרים מספקים. כל שנותר הוא להוכיח שיש קוד כזה. $\blacksquare$

טענה לכל $n \in \mathbb{N}$ קיים קוד מעל $\{0,1\}^N$ עם פרמטרים $\left(N, \frac{1}{100}, c, 2\right)$ כאשר $c \in [0,1]$ קבוע.

הוכחה: נראה אלג' שמוצא קוד שמוכל ב-$\{0,1\}^N$. נבחר $w_1 \in \{0,1\}^N$ ונשלול את כל מה שבכדור ברדיוס $\frac{1}{100}$ שלה. נבחר מילה נוספת זמינה ונשלול את מה שברדיוס שלה, וחוזר חלילה. מובטח לנו המרחק של לפחות $\frac{1}{100}$ בין כל שתי מילים. האלג' יפסיק כשאין עוד מילים זמינות.

ברור שלקוד מרחק $\frac{1}{100}$ לפחות. נוכיח שיש לקוד קצב קבוע. נניח שמצאנו k מילות קוד ואז נתקענו. מתקיים $2^N \leq k \cdot \left|B_0\left(\frac{1}{100}\right)\right|$ כי בכל פעם לכל היותר שללנו $\left|B_0\left(\frac{1}{100}\right)\right|$ מילים, ולכן

$$k \geq \frac{2^N}{\left|B_0\left(\frac{1}{100}\right)\right|} \geq \frac{2^N}{\binom{N}{\frac{N}{100}-1}} \overset{(*)}{\geq} 2^{c \cdot N}$$

$(*)$ מתקיים $\binom{N}{\alpha N} \approx 2^{N\left(\log_2\frac{1}{\alpha} + \log_2\frac{1}{1-\alpha}\right)}$.

לכן $R = \frac{\log k}{\log 2^N} = c$ כאשר c קבוע. $\blacksquare$