גישה היררכית לייצוג פרקטלי של אותות

חיבור על מחקר
לשם מילוי חלקי של הדרישות לקבלת התואר
מגיסטר למדעים
בהנדסת חשמל

מאת

צחי בהרב

הוגש לסנט הטכניון - מכון טכנולוגי לישראל

אדר תשנ"ד
חיפה
פברואר 1994

המחקר נעשה בהנחיית פרופ' דוד מלאך ודר' אהוד קרנין בפקולטה להנדסת חשמל - טכניון, חיפה.

דר' אהוד קרנין הוא עם .I.B.M ישראל.

ברצוני להודות לפרופ' דוד מלאך ולדר' אהוד קרנין על הנחייתם.

כמו כן, תודה לכל אנשי המעבדה לעיבוד אותות, ובפרט לזיוה אבני, על העזרה הטכנית הרבה.

תודה נוספת לאותם אלו שהשיחות איתם היפרו את העבודה, ובעיקר לשניים רחוקים (פיסית), דר' Geir O. Øien מנורבגיה ודר' Yuval Fisher מארה"ב.

תודה אחרונה לבוחנים, ולפרופ' משה ברגר בפרט, על הערותיהם, אשר תרמו לשיפור העבודה.

להורי, אביבה ויעקב בהרב

תוכן ענינים

תוכן עניינים (המשך)

תוכן עניינים (המשך)

תוכן עניינים (המשך)

רשימת איורים

רשימת איורים (המשך)

תקציר

בעבודה זו מתוארים תחילה עקרונות הפעולה הבסיסיים של מקודד פרקטלי של אותות. תהליך הקידוד מבוסס על חיפוש דמיון בין חלקים שונים באות. תוצאת המקודד היא מערכת פונקציות איטרטיבית (IFS-Iterated Function System), אשר מקרבת את האות כנקודת-שבת של טרנספורמציה מכווצת.

החלק המקורי של העבודה מתחיל בהצגת מבנה היררכי של קוד ה-IFS, אשר מקשר בין הקוד ובין נקודת-השבת ברזולוציות שונות. משפט הפונקציה המוכלת בקוד, המוכח בהמשך, מראה את קיומה של פונקציה של משתנה רציף, אשר קשורה ישירות לקוד. בעזרת פונקציה זו, הנקראת הפונקציה המוכלת בקוד, ניתן לחשב את נקודת-השבת של הקוד בכל רזולוציה רצויה.

לפני החלק הדן ביישומים מוצגת כתיבה מטריצית של קוד ה-IFS. כתיבה זו מאפשרת בהמשך ניסות משפט המתאר חסם על המימד הפרקטלי של הפונקציה המוכלת בקוד, וכמו כן מאפשרת הכתיבה המטריצית את מציאת מקדם הכיווץ של הטרנספורמציה כערך העצמי הגדול ביותר של מטריצה.

החלק הדן ביישומים פותח באפשרות פיענוח מהיר של קוד IFS. השיטה הרגילה לפענח קוד הינה בעזרת הפעלה חוזרת של מערכת פונקציות על אות שגודלו כגודל האות המקורי.

הייצוג ההירכי מאפשר לבצע את האיטרציות על האות ברזולוציה נמוכה, ואח"כ לחשב, ללא איטרציות, את האות ברזולוציה הגבוהה. יישום נוסף המוצג הוא האפשרות לביצוע אינטרפולציה פרקטלית של האות. אנו מוכיחים כי ניתן לחשב חסם (הדוק) על המימד הפרקטלי של הפונקציה המוכלת בקוד ישירות מתוך הקוד, ומתוך כך לקבל מידע על 'אופי' האינטרפולציה. לסיום חלק זה של יישומים, מתוארת השפעת אופן הדגימה של האות על דרך מציאת הקוד.

בפרק נפרד מתואר חסם משופר על שגיאת הקידוד. החסם המוצע מתחשב בכיווציות הטרנספורמציה, ובשגיאת הייצוג של האות ברזולוציות השונות. יתרונותיו על החסם הקלאסי נובעים מכך שהוא מאפשר עבודה נוחה גם עם טרנספורמציות שהן רק מכווצות לבסוף (eventually contractive), ומהיותו הדוק יותר.

לסיכום, מוצגות מסקנות ומספר כיווני המשך, הנובעים ישירות מתוך הייצוג ההירכי שתואר בעבודה.

רשימת סמלים וקיצורים

IFS (Iterated Functions System) - מערכת פונקציות איטרטיבית.

$\Re^N$ - מרחב N מימדי.

$d(\cdot,\cdot)$ - מטריקה.

$\cdots, c, b, a$ - סקלרים.

$\cdots, \vec{b}, \vec{a}$ - וקטורים.

$\cdots, [B], [A]$ - מטריצות.

$\vec{\mu}_o, \vec{u}$ - וקטור מקור.

$\vec{f}, \vec{x}_f$ - נקודת שבת.

N - גודל וקטור.

φ - פונקצית כיווץ מרחבי.

$\vec{r}, \vec{d}$ - בלוק-תחום (Domain-block) ובלוק-טווח (Range-block), בהתאמה.

B, D - גודל בלוק-תחום ובלוק-טווח, בהתאמה.

D_h - מרחק בין בלוקי תחום סמוכים.

M_R, M_D - מספר בלוקי-תחום ובלוקי טווח, בהתאמה.

s - גורם כיווץ.

a - גורם כיול.

b - גורם הסחה.

$G(\cdot)$ - הפונקציה המוכלת בקוד.

W - הטרנספורמציה המתארת את קוד ה-IFS.

$\mathcal{D}$ - מימד פרקטלי.

פרק 1

מבוא

חקר הפרקטלים החל כנושא מתמטי טהור הקשור לכאוס (למשל, פתית-השלג של וון-קוך (von-Koch snowflake), משולש סירפנסקי [19]). בשנת 1982 פירסם מנדלברוט את ספרו "The Fractal Geometry of Nature" [42], ובו טען שצורות טבעיות רבות הן פרקטליות. בין הצורות הפרקטליות השונות שאותן הוא מונה, ניתן להזכיר את קו-החוף, עננים בשמיים, קו-רקיע של הרים ועוד.

השימוש בפרקטלים לצרכים הנדסיים נעשה פופולרי עם התפתחות היכולת הגרפית של מחשבים, שאיפשרה ליצור ולראות, בקלות יחסית, צורות פרקטליות. התמונות שנוצרו, למשל עננים, היו בעלות מראה טבעי, והוו תמריץ להתפתחות נושא ההרכבה של תמונות פרקטליות (picture synthesis) [53].

עם התפתחות יצירת התמונות הפרקטליות, הוחל גם בנסיונות להשתמש במודל פרקטלי של הטבע כדי לייצג תמונות. למשל, אפיון מרקמים שונים בתמונות בעזרת המימד הפרקטלי שלהם [62, 56, 15]. כמו כן החלו נסיונות שונים להשתמש במודל הפרקטלי לצורך דחיסת תמונות [71, 55]. בעבודתו של Pentland [55] למשל, נעשה נסיון להשתמש בתכונה של דמיון

עצמי בין רזולוציות שונות של אות פרקטלי, כדי להביע יתירות במבנה פירמידלי של האות.

אחת השיטות ליצירת תמונות-פרקטליות היא בעזרת מערכת איטרטיבית של פונקציות (Iterated Function Systems) [52]. כיוון שה-IFS התגלה ככלי מוצלח בהרכבת תמונות, נעשו נסיונות להשתמש ב-IFS לצורך קידוד תמונה נתונה. כלומר, בהינתן תמונה, נמצא מהו ה-IFS שמתאר אותה [7], ואותו נשמור כקוד עבור התמונה. אם ה-IFS שנמצא מתאר תמונה קרובה בלבד לתמונה הנתונה, הרי הקוד שהתקבל הוא קוד עם עיוות (lossy-code). הבעיה העיקרית עם נסיונות ראשוניים אלו היתה העובדה שמציאת ה-IFS המתאים נעשתה באופן ידני: המשתמש היה צריך לשבת מול התמונה, ולנסות למצא באופן נסיוני את הקוד המתאים.

את המעבר לגישה שיטתית למציאת הקוד הציע לראשונה Jacquin [32, 33]. גישה זו מתבססת על הגבלת מספר הפונקציות המותרות, ולכן חיפוש הפונקציה המתאימה ניתן להעשות באופן שיטתי ע"י מכונה. מובן שהגבלת מספר הפונקציות גרמה לכך שישנן תמונות אשר אינן ניתנות לתאור מדויק בעזרת פונקציות אלו, ולכן הקידוד עבורן הוא בהכרח קידוד עם עיוות (lossy-coding). אולם, רובן המכריע של העבודות בדחיסת תמונות בטכניקות פרקטליות מאז התפרסם המאמר של Jacquin, מתבסס על גישתו, ומנסה להציע לה שיפורים [21, 48, 70, 46]. שיפורים אלה יתוארו בהמשך (פרק 4).

1.1 הצגת הבעיה

המעבר מנושא מתמטי מופשט (כאוס ופרקטלים), אל שיטה הנדסית המבוססת על מערכות איטרטיביות של פונקציות, גרם לכך שמספר תכונות בסיסיות, האופייניות לפרקטלים, כבר לא באו לידי ביטוי. לפיכך, במקום 'קידוד פרקטלי של תמונות', החלו להופיע שמות שונים,

כמו 'קידוד בעזרת IFS' או 'קידוד בעזרת נקודת-שבת'.

לדוגמא, תכונה שכמו 'אבדה' במעבר לעולם הדיסקרטי, היא תכונת הדימיון בין רמות רזולוציה שונות של האות. תכונה זו, שהיא תכונה בולטת ובסיסית בפרקטלים, אינה באה לידי ביטוי בקידוד בעזרת IFS.

כמו כן, בעוד שרוב התורה המתמטית הקשורה לפרקטלים [5, 3] עסקה במידות ובמבנה כללי של IFS, הרי שהאותות שאותם ניסו לקודד\ליצג היו דיסקרטיים, ומבנה ה-IFS הוגבל לסוג מסוים בלבד. בהתאם לכך, לא היתה הרחבה נוחה של תכונות ומשפטים אל העולם הדיסקרטי ואל המקרה של טרנספומציות מוגבלות.

1.2 סיכום התרומות העיקריות של העבודה

התרומה העיקרית של עבודה זו היא בהצגת קשר בין רמות רזולציה שונות של אות אשר קודד בעזרת מערכת איטרטיבית של פונקציות (IFS). הקשר בין הרמות השונות הוא ביטוי לתכונת הדמיון העצמי (self-similarity) של פרקטלים. מתוך הבנת הקשר ההירכי נובעות מספר תוצאות, אשר אף הן מובאות בעבודה זו.

להלן מובאות התרומות העיקריות של העבודה לפי סדר ענייני (כלומר, מהכלל אל הפרט, ומתכונה בסיסית אל שימוש שלה), אך לא בהכרח לפי סדר חשיבות:

- הוכחת קשר בין האות ברזולוציות שונות ובין הקוד שלו, ובהתאם לכך ייצוג הירכי של האות. כלומר, בהינתן האות ברזולוציה מסוימת, ובהינתן הקוד המיצג אותו, ניתן לחשב ישירות את האות ברזולציה אחרת (גם גבוהה יותר!).
- הוכחת קיום פונקציה מוכלת בקוד (IFS embedded function), אשר היא פונקציה של משתנה רציף, שמתוכה ניתן לגזור את האות בכל רזולוציה דרושה. ניתן להתיחס

לפונקציה זו כמקרה גבול, של רזולוציה אינסופית.

- פענות מהיר של הקוד. שיטת הפענות המוצעת נובעת ישירות מהייצוג ההירכי של האות, אשר מוכח בעבודה זו. שיטה זו מביאה לחיסכון של סדר גודל במספר החישובים הדרוש לפענות.

- אינטרפולציה פרקטלית של האות. בהינתן האות ברזולוציה מסוימת, ובהינתן הקוד שלו, ניתן, כאמור, למצוא את האות ברזולציות גבוהות יותר. בעבודה אף דנים בתכונות מסוימות של האינטרפולציה.

- חסם על הממד פרקטלי של הפונקציה המוכלת בקוד. בעבודה מוצג משפט המקשר חסם על הממד הפרקטלי של הפונקציה המוכלת בקוד עם הקוד עצמו (או ליתר דיוק, עם הייצוג המטריצי של הקוד, שאף הוא מוצג בעבודה).

- חסם משופר לצורך קידוד. כפי שנראה בהמשך, המוטיבציה לאלגוריתם הקידוד המקובל נובעת מחסם על שגיאת הקידוד כפונקציה של "שגיאת הפסיפס" (Collage-bound). חסם זה מתבסס על האות ברזולציה נתונה. בעבודה זו אנו מציגים חסם משופר, המתבסס על האות ברזולציות שונות. חסם זה יותר הדוק, ומוביל לאלגוריתם קידוד שונה, שיכול להוביל לשגיאת קידוד קטנה יותר.

- שיטות דגימה שונות. ההנחה שמונחת בבסיס שיטת הקידוד היא שהאות (תמונה) הרציף הוא פרקטל. יחד עם זאת, האות שאותו מנסים לקודד הוא אות דיסקרטי, שהתקבל ע"י דגימה של האות הרציף. בעבודה מוצגות דרכים למציאת הקוד עבור אות דיסקרטי, תוך התחשבות בשיטת הדגימה. דרכים אלה למציאת הקוד מאפשרות קבלת קוד טוב יותר מזה המתקבל בשיטה הרגילה, אשר אינה לוקחת בחשבון את שיטת הדגימה.

כמו כן, עבודה זו פותחת אפשרויות רבות לעבודה תאורטית, וכמובן לשימושים מעשיים, כפי שיפורט בסופה.

1.3 מבנה העבודה

פרק 2 פותח במספר מושגים מתמטיים הדרושים להמשך העבודה. לאחר מכן מוצגות באופן מפורש שיטות הקידוד והפענות של אות בעזרת מערכת איטרטיבית של פונקציות (IFS). מוגדרת גם תכולת הקוד המתאר אות מסוים, ומוצגת האפשרות לפענח קוד נתון ברזולוציות שונות.

פרק 3 פותח את החלק המקורי של עבודה זו. ע"פי המוצג בפרק 2, ניתן לפענח קוד נתון במספר רזולוציות שונות. רזולוציות שונות אלה יוצרות מבנה הירכי, המבוסס על האות המקודד. לגבי מבנה זה, מוכחים שני משפטים: האחד קושר את האות ברזולציות השונות והקוד, והשני קובע כי קיימת פונקציה (יחידה !) אשר בעזרתה ניתן לחשב ישירות כל רזולוציה של האות. שני משפטים אלה מהווים את הבסיס להמשך העבודה.

פרק 4 מורכב משני חלקים. החלק הראשון מתאר נסוח מטריצי של הקוד. נסוח זה של הקוד מאפשר כתיבה של משפט המתאר חסם על המימד הפרקטלי של הפונקציה המוכלת בקוד (מתואר בפרק 5), וכן מהווה נסוח יעיל לצורך מציאת כיווציות הקוד .חלקו השני של הפרק עוסק בהרחבות שונות לשיטת הקידוד הבסיסית ,ובפרט מתוארת הרחבה שנזדקק לה בהמשך.

פרק 5 עוסק ביישומים שונים של התאוריה שהוצגה בפרק 3, ושל הנסוח המטריצי שהוצג בפרק הקודם. הפרק פותח ביישום המבנה ההירכי לצרכי פענוח מהיר של הקוד. לאחר מכן מתואר יישום לשם השגה ישירה של רזולוציה גבוהה יותר מרזולוצית האות

המקורי שעבורו נמצא קוד ה-IFS. יישום נוסף המוצג בפירוט הוא היישום למקרה של פונקציות דגימה שונות: כלומר, בהנחה שהאות הדיסקרטי שבידינו התקבל מאות רציף ע"י דגימה, אנו מתארים מהי הדרך הטובה ביותר למציאת הקוד המתאר את האות.

פרק 6 מציג חסם משופר על שגיאת הקידוד. חסם זה הדוק יותר מהחסם הרגיל, ומנצל את המבנה ההירכי שתואר בעבודה.

פרק 7 מסכם את העבודה, ומציג בעיות-פתוחות וכיוונים להמשך.

בנספחים מופיעות הוכחות של המשפטים המקוריים של עבודה זו, וכן דוגמאות לאינטרפולציה פרקטלית. בנספחים (א) ו-(ג) מופיעות שתי הוכחות שונות של משפט הקשר ההירכי. החשיבות של ההוכחה בדרכים שונות מוסברת שם. בנספת (ד) מוצגות תמונות, המדגימות ביצוע אלגוריתם לרזולוצית-על, המוצע בפרק (5).

פרק 2

מושגים והגדרות במערכת איטרטיבית של פונקציות

2.1 רקע מתמטי

בסעיף זה יובאו מספר הגדרות ומשפטים, אשר יידרשו להמשך. מושגים בסיסיים יותר הקשורים במרחבי בנך (למשל מרחב לינארי, נורמה), לא יובאו פה, וניתן למצוא אותן בכל ספר בסיסי באנליזה [61].

יהיה X מרחב מטרי עם מטריקה d. הטרנספורמציה $T: X \rightarrow X$ נקראת טרנספורמציה מכווצת (contraction) אם

$$\exists\, s \in [0,1) \mid \ \forall\, x,y \in X \quad d(T(x),T(y)) \leq s\, d(x,y). \tag{2.1}$$

המספר s הנמוך ביותר המקיים את המשוואה, נקרא גורם הכיווץ של הטרנספורמציה.

אם X הוא מרחב מטרי שלם, אזי משפט הפונקציה המכווצת (contraction mapping theorem) [4] מבטיח את קיום התכונות הבאות:

- קיימת נקודה אחת ויחידה, $x_f \in X$, כך שמתקיים $T(x_f) = x_f$. נקודה זו נקראת

נקודת-שבת של הטרנספורמציה T.

- עבור כל נקודה $x\in X$, הסידרה $\{T^{\circ n}(x) : n = 0, 1, \ldots\}$ מתכנסת לנקודה x_f, כלומר $.lim_{n\to\infty} T^{\circ n}(x) = x_f \quad ,\forall x\in X$

כלומר, נקודת השבת x_f מוגדרת חד-ערכית ע"י T, וניתן למצוא אותה ע"י איטרציות. בהמשך, כפי שנראה, המרחב X יסמל את מרחב התמונות, x_f היא תמונה שצריכה להיות קרובה ככל האפשר לתמונה שאותה אנו רוצים לקודד, ו-T היא מערכת הפונקציות הנשמרת כקוד, כלומר ה-IFS עצמו.

משפט נוסף החשוב מאד להמשך, הוא משפט הפסיפס (Collage-theorem) [4].

משפט 1 (Collage-theorem) יהי (X, d) מרחב מטרי שלם. תהי T טרנספורמציה מכווצת ב-X, עם גורם כיווץ $s < 1$. תהי x_f נקודת השבת של T, כלומר $T(x_f) = x_f$. אזי,

$$.\forall x, y\in X, \quad d(y, x_f) \leq \frac{1}{1-s} d(y, T(y)) \tag{2.2}$$

לנקודה $T(y)$ קוראים תמונת הפסיפס של y (Collage).

כיוון שהוכחת משפט זה מבהירה מספר תכונות הדרושות להמשך, היא מובאת כאן בקיצור. בהוכחה נעשה שימוש חוזר באי-שיוויון המשולש ובתכונות פונקציה מכווצת.

$$\begin{aligned}
d(y, x_f) &\leq d(y, T(y)) + d(T(y), x_f) \\
&= d(y, T(y)) + d(T(y), T(x_f)) \\
&\leq d(y, T(y)) + s \cdot d(y, x_f) \\
&\leq d(y, T(y)) + s \cdot d(y, T(y)) + s \cdot d(T(y), x_f) \\
&\leq d(y, T(y)) + s \cdot d(y, T(y)) + s^2 \cdot d(y, x_f) \\
&\leq \ldots \leq d(y, T(y)) \cdot (1 + s + s^2 + \ldots + s^n) + s^{n+1} \cdot d(y, x_f)
\end{aligned}$$

וכיוון ש-s קטן מאחד, מתקבל המשפט.

מבלי לפרט, נציין מספר הערות לגבי המשפטים וההגדרות.

- טרנספורמציה $T(\cdot)$ אשר אינה מכווצת, אולם קיים m כך ש-$T^m(\cdot)$ היא טרנספורמציה מכווצת, נקראת טרנספורמציה מכווצת לבסוף (eventually contractive). קיימת הרחבה למשפט הפסיפס (collage-theorem) עבור המקרה של פונקציה מכווצת לבסוף [21] [50].
- בהגדרת הכיווציות קיימת תלות ברורה בבחירת המטריקה. עבור מטריקות שונות, נקבל גורם כיווץ שונה. יחד עם זאת, אם פונקציה היא מכווצת במטריקה אחת, אזי היא מכווצת-לבסוף (eventually-contractive) בכל מטריקה שקולה לה (ראה מיד), ולכן גם בעלת נקודת שבת במטריקה השקולה.

מטריקה שקולה: יהיו d_1 ו-d_2 שתי מטריקות במרחב X. המטריקות הן שקולות אם קיימים קבועים a, b חיוביים ממש כך שמתקיים

$$\forall x, y \in X \quad a \cdot d_1(x,y) \leq d_2(x,y) \leq b \cdot d_1(x,y) \quad .$$

2.1.1 מודל האות

בהמשך נעסוק בעיקר באותות חד מימדיים, שייוצגו ע"י וקטורים (ולעתים יקראו בלוקים). הרחבות למקרה של אותות דו מימדיים, למשל תמונות, הן מידיות ברוב המקרים, וחלק מההרחבות אף ניתן בחלק העבודה הדן ביישומים. אנו נשתמש בסימונים הבאים: וקטורים יסומנו באותיות קטנות עם חץ (כמו $\vec{a}$), ומטריצות יסומנו באותיות גדולות עם סוגריים מרובעות ($[A]$). סקלרים יסומנו באותיות רגילות (כמו a).

בהמשך נעסוק במרחב מטרי שלם, $(\Re^N, d^\infty)$, כאשר:

1. $\Re^N$ הוא מרחב N-מימדי. כל נקודה במרחב $\Re^N$ היא וקטור באורך N של מספרים ממשיים. לכן,

$$\vec{x} \in \Re^N \quad \Leftarrow \quad \vec{x} = (x_1, x_2, \cdots, x_i, \cdots, x_N)^T. \tag{2.3}$$

2. d^∞ היא מטריקת המקסימום, המוגדרת:

$$d^\infty(\vec{x}, \vec{y}) = \max_{i=1,\cdots,N} |x_i - y_i| \qquad \vec{x}, \vec{y} \in \Re^N. \tag{2.4}$$

הערה: נגדיר את המטריקה d_2,

$$d_2(\vec{x}, \vec{y}) = \left\{ \sum_{i=1}^{N} (x_i - y_i)^2 \right\}^{\frac{1}{2}} \qquad \vec{x}, \vec{y} \in \Re^N.$$

עבור $\Re^N$ המטריקות d_2 ו-d^∞ הן שקולות, כיוון שקיים:

$$1 \cdot d^\infty(\vec{x}, \vec{y}) \leq d_2(\vec{x}, \vec{y}) \leq \sqrt{N} \cdot d^\infty(\vec{x}, \vec{y}) \quad .$$

2.2 קידוד/פיענוח בעזרת מערכת איטרטיבית של פונקציות מקומיות (IFS)

2.2.1 קידוד

המשימה של מציאת קוד IFS עבור אות $\vec{\mu}_o$ מורכבת ממציאת טרנספורמציה W, כך שנקודת השבת של הטרנספורמציה תהיה קרובה ככל האפשר ל-$\vec{\mu}_o$. בסעיף זה נפרט את תהליך מציאת הקוד.

האות (וקטור) שאותו יש לקודד הוא $\vec{\mu}_o \in \Re^N$. אנו מחפשים טרנספורמציה W, כך שתתקיימנה ארבע הדרישות הבאות:

1. W ממפה את המרחב לתוך עצמו.

$$\begin{aligned} W \ : \ \Re^N &\to \Re^N \\ \vec{v} &\mapsto \vec{u} = W(\vec{v}) \end{aligned} \qquad (2.5)$$

2. W היא טרנספורמציה מכווצת.

$$\exists\, s \in [0,1) \ \mid \ \forall\, \vec{x}, \vec{y} \in \Re^N \ , \quad d^\infty(W(\vec{x}), W(\vec{y})) \le s\, d^\infty(\vec{x}, \vec{y}). \qquad (2.6)$$

3. כפי שנראה מיד, W היא הקוד המתאר את $\vec{\mu}_o$. מכיון שכך, הרי הדרישה השלישית היא שתאורה (של הטרנספורמציה) יהיה 'פשוט' יותר מתאור אות המקור, $\vec{\mu}_o$. המילה 'פשוט' הושמה במרכאות, משום שהיא אינה מוגדרת היטב בשלב זה. יחד עם זאת, משמעותה המעשית היא הגבלת מספר הטרנספורמציות W המותרות למספר סופי, כך שבעזרת מספר סיביות מצומצם נוכל לתאר איזה מהן נבחרה לשמש כקוד.

שלוש הדרישות הראשונות מגדירות את אוסף הטרנספורמציות המותרות, $W \in \mathcal{W}$.

4. כיוון ש-W היא טרנספורמציה מכווצת במרחב מטרי שלם, הרי שיש לה נקודת שבת אחת ויחידה $\vec{f}_w \in \Re^N$ המקיימת $\vec{f}_w = W(\vec{f}_w)$. הדרישה הרביעית היא ש-W תהיה טרנספורמציה שמביאה למינימום את המרחק בין $\vec{\mu}_o$ ובין $\vec{f}_w$:

$$W = arg \min_{W \in \{\mathcal{W}\}} d^\infty(\vec{\mu}_o, \vec{f}_w) \qquad (2.7)$$

כאשר לכל W קשורה $\vec{f}_w$ אחת ויחידה.

המשימה של מציאת טרנספורמציה מתאימה W היא משימה מורכבת מאד, משום שהיא מחייבת מינימיזציה על פני מספר רב של טרנספורמציות. כדי לאפשר פתרון שיטתי של הבעיה, הוצע להגביל את אוסף הטרנספורמציות המותרות (סעיף 3 לעיל) לסוג מסוים של טרנספורמציות [32]. W אשר תמצא בדרך זו אינה בהכרח האופטימלית, אולם זוהי פשרה שיש לעשות כדי לפתור את הבעיה.

נגביל אפוא את הטרנספורמציות המותרות W, $\vec{u} = W(\vec{v})$, להיות מערכת המורכבת מ-M_R פונקציות w_i. כל w_i מוגבלת אף היא להיות מהצורה:

$$\begin{aligned} w_i \quad : \quad \Re^D \quad &\rightarrow \quad \Re^B \\ \vec{d}_{m_i} \quad &\mapsto \quad \vec{r}_i = w_i(\vec{d}_{m_i}) = a_i \varphi(\vec{d}_{m_i}) + b_i \vec{1}_B \end{aligned} \qquad (2.8)$$

כאשר:

$\vec{d}$ - זהו בלוק-תחום (*domain-block*), והוא בגודל D. $\vec{d}_{m_i}$ הוא אפוא בלוק-תחום ה-m_i-י, שהוא פשוט D איברים רצופים מתוך $\vec{v}$. השימוש בסימון m_i בא להדגיש את העובדה שבלוק-תחום $\vec{d}_{m_i}$ ממופה ל-$\vec{r}_i$. ביטוי מפורש לקבלת בלוק התחום ה-$\vec{d}_{m_i}$ הוא:

$$\vec{d}_{m_i}(j) = \vec{v}((m_i - 1)D_h + j) \quad , \qquad j = 1, 2, \cdots, D$$

כאשר D_h מוגדר כהזזה בין שני בלוקי-תחום סמוכים, והוא נתון מפורשות בקוד.

- $\vec{r}$ - זהו בלוק-טווח (*range-block*) , והוא בגודל B ‏($B < D$). ‏$\vec{r}_i$ הוא אפוא בלוק-טווח ה-i-י, וכמובן $\vec{r}_i \in \vec{u}$.
- φ - פונקצית כיווץ מרחבי (*Spatial contraction* function), אשר ממפה בלוקים בגודל D לבלוקים בגודל B.
- a_i - גורם ביול **סקלרי** (scalar *scaling* factor),

$$.a_i \in \Re, \qquad |a_i| < 1 \tag{2.9}$$

- b_i - גורם הסחה **סקלרי** (scalar *offset* value), $b_i \in \Re$.
- $\vec{1}_B$ - וקטור בגודל B שכולו 1-ים.

שלושת הפרמטרים (a_i, b_i, m_i) נקראים פרמטרי הטרנספורמציה.

אם נשתמש בסימון האיחוד לסמן שירשור של בלוקים, אזי נוכל לכתוב:

$$\begin{aligned} \vec{u} &= W(\vec{v}) \\ W(\vec{v}) &= \bigcup_{i=1}^{M_R} w_i(\vec{d}_{m_i}), \qquad \vec{d}_{m_i} \in \vec{v} \end{aligned} \tag{2.10}$$

וכמו כן, לפי (2.8), ניתן לכתוב:

$$.\vec{u} = \bigcup_{i=1}^{M_R} \vec{r}_i, \qquad \vec{r}_i \in \vec{u} \tag{2.11}$$

אורכו של $\vec{u}$, המתקבל משירשור של M_R בלוקי-טווח באורך B כל אחד, הוא

$$.N = M_R \cdot B \tag{2.12}$$

שירשור בלוקי-הטווח יכול גם להיכתב באופן הבא:

$$\vec{u}((i-1)\cdot B+j)=\vec{r}_i(j) \quad ; \quad i=(1,\cdots,M_R), \qquad j=(1,\cdots,B) \qquad (2.13)$$.

כעת, בהנחה שכל הפרמטרים המתארים את W נתונים, ניתן לתאר את חישוב $\vec{u}=W(\vec{v})$ באופן הבא:

סיכום 1 : אלגוריתם לביצוע הטרנספורמציה $\vec{u}=W(\vec{v})$

Summary 1 : Algorithm for Performing the Transformation $\vec{u}=W(\vec{v})$

1. עבור $i=1,\ldots,M_R$
   - (א) הפק את בלוק–התחום $\vec{d}_{m_i}$ מתוך הוקטור $\vec{v}$.
   - (ב) חשב את בלוק–הטווח

$$\vec{r}_i=w_i(\vec{d}_{m_i})=a_i\varphi(\vec{d}_{m_i})+b_i\vec{1}_B \qquad (2.14)$$.

2. שרשר את בלוקי–הטווח, $\vec{r}_i$, $i=(1,\cdots,M_R)$, לפי סדר ה–i, כדי לקבל וקטור חדש, $\vec{u}$. אורך הוקטור $\vec{u}$ הוא : $N=M_R\cdot B$.

W כפי שתואר לעיל נקרא מערכת (איטרטיבית של) פונקציות מקומיות (Local IFS) [8] (במאמרים קודמים נעשה שימוש גם בביטוי *block-wise* transformation [2]). התואר 'איטר-

טיבית' נוסף בסוגריים להגדרת המערכת, משום שמפעילים את המערכת שוב ושוב, ואינו קשור לעצם הגדרת המרכיבים של המערכת. שם התואר 'מקומיות' בא להדגיש את העובדה שכל אחת מהפונקציות במערכת פועלת על אזור מקומי מתוך האות (על בלוק התחום). זאת, בניגוד למערכות פונקציות שבהן כל פונקציה ממפה את כל האות לתחום מסוים. בהמשך נשמיט את שם התואר 'מקומיות', אך נזכור תמיד שכל הדיון הוא על סוג זה של טרנספורמציות.

עד כה תיאור ה-w_i-ים היה כללי. כעת נטיל הגבלות נוספות ונניח מספר הנחות על הפרמטרים השונים, כך שהדיון יהיה הן יותר מעשי והן יותר נות.

סיכום 2 : הנחות והגבלות על פרמטרים

Summary 2 : Parameter Restrictions and Assumptions

1. N – אורך אות המקור, הוקטור שאותו מקודדים, הוא חזקה שלמה של 2.
2. $B = 2^r$ – אורך בלוק–הטווח. B הוא אפוא גם כן חזקה שלמה של 2.
3. $D = 2B$ – אורך בלוק–התחום.
4. $D_h = B$ – D_h מוגדר כהזזה בין שני בלוקי–תחום סמוכים. לכן, מספר בלוקי–התחום הוא $M_D \triangleq (\frac{N-D}{D_h}+1)$, וכל אחד נתון מפורשות ע"י

$$\vec{d}_{m_i}(j) = \vec{v}((m_i-1)D_h+j) \quad ; \qquad (2.15)$$

$$m_i = 1, 2, \cdots, M_D \quad ; \quad j = 1, 2, \cdots, D.$$

בלוקי–תחום סמוכים חופפים לפי הנחות אלו, משום ש–$D_h < D$.

5. $\varphi(\cdot)$ – פונקצית הכיווץ המרחבי מוגדרת:

$$\varphi(\vec{d}_{m_i})(j) \triangleq \frac{1}{2}(\vec{d}_{m_i}(2j) + \vec{d}_{m_i}(2j-1)), \qquad j = 1, 2, \cdots, B \quad (2.16)$$

כלומר, $\varphi(\cdot)$ מכווצת בלוקים באורך $D = 2B$ לבלוקים באורך B, ע"י מיצוע זוגות של איברים סמוכים ב–$\vec{d}_{m_i}$.

כעת ניתן לסכם את תוכנו של קוד IFS עבור אות מסוים. תוכן הקוד הם הפרמטרים המתארים את W, ראה (4.24) והוא מתואר בטבלה הבאה:

סיכום 3 : קוד IFS

Summary 3 : IFS-code

1. B – אורך בלוקי–טווח.
2. M_R – מספר בלוקי–טווח.
3. M_R שלשות של פרמטרי–טרנספורמציה (a_i, b_i, m_i).

כל הנתונים האחרים הדרושים לפענות, כמו $D = 2B$, $D_h = B$, $N = M_R B$, ואחרים, ניתנים לחישוב מתוך הקוד תוך שימוש בהנחות שתוארו קודם.

נתאר כעת את תהליך הקידוד, כלומר תהליך מציאת M_R השלשות (a_i, b_i, m_i), המרכיבות את הקוד. כפי שנאמר, המטרה היא להביא למינימום את $d^\infty(\vec{\mu}_o, \vec{f})$, כאשר $\vec{\mu}_o$ הוא האות (וקטור) המקורי, ו-$\vec{f}$ היא נקודת השבת של הטרנספורמציה (אותה אנו מחפשים).

כיוון שלפי משפט הפסיפס (Collage theorem) חסם עליון למרחק נתון לפי

$$d^\infty(\vec{\mu}_o, \vec{f}) \leq \frac{1}{1-s} d^\infty(\vec{\mu}_o, W(\vec{\mu}_o)), \tag{2.17}$$

מנסים להביא למינימום את $d^\infty(\vec{\mu}_o, W(\vec{\mu}_o))$ (שים לב שהגורם $\frac{1}{1-s}$ אינו נלקח בחשבון, אף-על-פי שהוא תלוי בבירור ב-W). שיטה זו, של חיפוש מינימום לחלק מהחסם העליון, לא תביא בהכרח למינימום של $d^\infty(\vec{\mu}_o, \vec{f})$, אולם זוהי כיום השיטה המעשית ביותר למציאת קוד IFS.

כיוון ש-W פועלת בנפרד על בלוקים שונים (blockwise transformation), ניתן לבצע את המינימיזציה בשלבים, כפי שיתואר להלן. תהליך המינימזציה מורכב ממציאת W אשר מקיימת $W(\vec{\mu}_o) \cong \vec{\mu}_o$. כלומר $\vec{\mu}_o$ היא בקרוב נקודת השבת של W. כיוון ש-W מגדירה חד-ערכית את נקודת השבת שלה, שמירת W, כלומר הפרמטרים המגדירים אותה, מגדירה קוד (עם עיוות) ל-$\vec{\mu}_o$.

כדאי לשים לב שבקשר $\vec{\mu}_o \cong W(\vec{\mu}_o)$, גם המשתנה וגם (בקירוב) התוצאה של W הם $\vec{\mu}_o$. לכן, בתהליך הקידוד, $\vec{d}_{m_i} \in \vec{\mu}_o$ וכן $\vec{r}_i \in \vec{\mu}_o$.

סיכום 4 : קידוד IFS של $\vec{\mu}_o$

Summary 4 : IFS-Encoding of $\vec{\mu}_o$

1. שמור B בקובץ הקוד.
2. שמור M_R בקובץ הקוד, כאשר $M_R = N/B$, ו-N הוא האורך של $\vec{\mu}_o$.
3. חלק את $\vec{\mu}_o$ ל-M_R בלוקי-טווח,

$$\vec{r}_i(j) = \vec{\mu}_o((i-1)\cdot B + j), \qquad (2.18)$$

$$i = (1, \cdots, M_R), \qquad j = (1, \cdots, B).$$

4. הפק מתוך $\vec{\mu}_o$ את $M_D = (\frac{N-D}{D_h} + 1)$ בלוקי-התחום,

$$\vec{d}_l(j) = \vec{\mu}_o((l-1)D_h + j), \qquad (2.19)$$

$$l = 1, 2, \cdots, M_D, \qquad j = 1, 2, \cdots, D.$$

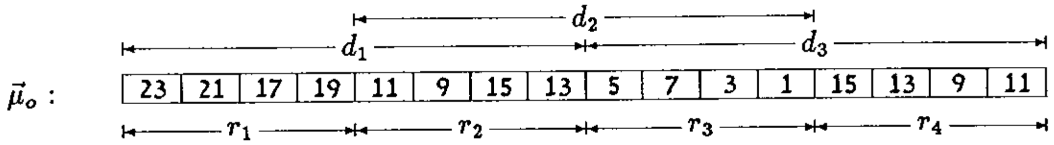
5. עבור $i = 1, \ldots, M_R$:

(א) מצא את הפרמטרים האופטימליים (a_i, b_i, m_i), כלומר את אלה שיביאו למינימום את

$$.d^{\infty}(\vec{r}_i, (a_i\varphi(\vec{d}_{m_i}) + b_i\vec{1}_B)) \qquad (2.20)$$

(ב) שמור את הפרמטרים (a_i, b_i, m_i) בקובץ הקוד.

המושגים והאלגוריתמים שתוארו עד כה, מודגמים בדוגמא הבאה. איור 2.1 (א)-(ב) מציג את הוקטור $\vec{\mu}_o$ ואת קוד ה-IFS שלו. הקוד נתון בצורת טבלה. ע"י ביצוע של הטרנספורמציה המתוארת בקוד על $\vec{\mu}_o$, ניתן לוודא שאכן $\vec{\mu}_o$ בדוגמא זו הוא נקודת שבת של הטרנספורמציה (כלומר, $\vec{\mu}_o = W(\vec{\mu}_o)$), ולכן הקידוד במקרה זה הוא ללא עיוות.



(א)

הסחה	בלוק-תחום מס'	כיול	בלוק-טווח מס'
b_i	m_i	a_i	i
12	1	0.5	1
8	3	0.5	2
0	2	0.5	3
4	1	0.5	4

$B = 4 \ , \ M_R = 4$

(ב)

$$\begin{aligned} \vec{r}_1 &= 0.5 \cdot \varphi(\vec{d}_1) + 12 = 0.5 \cdot \varphi([23, 21, 17, 19, 11, 9, 15, 13]) + 12 = \\ & 0.5 \cdot [22, 18, 10, 14] + 12 = [23, 21, 17, 19] \end{aligned}$$

(ג)

איור 2.1: קידוד IFS - (א) וקטור מקור $\vec{\mu}_o$ (ב) קוד IFS של וקטור מקור (ג) דוגמא לחישוב בלוק-טווח $\vec{r}_1$ תוך שימוש בשורת הקוד הראשונה מ-(ב) וכאשר בלוק-תחום הוא $\vec{d}_1 \in \vec{\mu}_o$, כפי שנתון ב-(א).

Fig. 2.1: IFS coding - (a) An original vector $\vec{\mu}_o$ (b) IFS code of $\vec{\mu}_o$ (c) Example of computing the first code-line on $\vec{\mu}_o$

2.2.2 פיענוח

תהליך הפיענוח מורכב ממציאת נקודת השבת של טרנספורמציה מכווצת W. הפיענוח מתבצע ע"י הפעלה חוזרת (איטרציות) של W על וקטור התחלתי כלשהו, עד אשר מגיעים למרחק מספיק קטן מנקודת השבת [4].

באיור 2.2 מודגם הפיענוח של הקוד הנתון באיור 2.1 (ב), כאשר וקטור ההתחלה הוא וקטור שכולו אפסים. באיור מודגמים הוקטורים המתקבלים אחרי הפעלה חוזרת של W. כמו כן, מודגמת גם קבלת בלוק-הטווח הראשון, באיטרציה הראשונה. מתוך כך ניתן לראות, שאם הוקטור ההתחלתי הוא וקטור שכולו אפסים, הרי שבאיטרציה הראשונה הערך שמקבל כל בלוק-תחום הוא ערך ההסחה שנתון בקוד עבורו.

וקטור																איטרציה
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	4	4	4	0	0	0	0	8	8	8	8	12	12	12	12	1
8	8	10	10	0	0	4	4	10	10	8	8	16	16	18	18	2
9	8	12	13	0	2	5	4	12	13	8	10	17	16	20	21	3

(א)

$$0.5 \cdot \varphi(\vec{d_1}) + 12 =$$

$$0.5 \cdot \varphi([0,0,0,0,0,0,0,0]) + 12 =$$

$$0.5 \cdot [0,0,0,0] + 12 =$$

$$[12,12,12,12]$$

(ב)

איור 2.2: פיענוח ע"י איטרציות - (א) תוצאות 3-האיטרציות הראשונות (ב) דוגמא לחישוב בלוק-טווח ראשון באיטרציה ראשונה.

Fig. 2.2: Decoding by iterations - (a) Results of first 3-iterations (b) Demonstration of the computation of the first range-block in the first iteration.

בהמשך לדוגמא, נדגים כעת עובדה חשובה הקשורה לפיענות (ולהגדרת קוד ה-IFS בכללו). בדוגמא לעיל, קוד ה-IFS, עם גודל הבלוק הנתון $B = B_1 = 4$, תיאר טרנספורמציה הפועלת על וקטורים באורך $N = M_R \cdot B = 4 \cdot 4 = 16$, כלומר טרנספורמציה $W^1 : \Re^{16} \mapsto \Re^{16}$, עם נקודת שבת $\vec{f}^1 \in \Re^{16}$. אולם, נניח שהערך של B שונה לערך אחר מזה הנתון בקוד, למשל $B = \frac{1}{2}B_1 = 2$. ע"י כך נוצרה טרנספורמציה חדשה, שנסמנה כ- $W^{\frac{1}{2}}$, שהיא טרנספורמציה מכווצת ב-$\Re^8$. תהליך הפיענות של הקוד החדש, עם $W^{\frac{1}{2}}$, יביא אותנו לנקודת שבת חדשה שהיא וקטור $\vec{f}^{\frac{1}{2}}$ באורך $N = N_{\frac{1}{2}} = 8$, שהוא חצי מאורכו של $\vec{f}^1$. מכאן, שקוד ה-IFS ניתן לפיענות במרחבים שונים, ובכל מרחב הוא מתאר נקודת שבת. מכאן והלאה ניצמד לסימון שלפיו גודל בלוק-טווח $B = B_1$ מוביל לנקודת שבת $\vec{f}^1$, ואילו $B = pB_1$ מוביל ל-$\vec{f}^p$. באיור 2.3 (א)-(ג) מתוארות 3 נקודות שבת של אותו קוד (תוך שימוש ב-B-ים שונים), וכל נקודת שבת היא במרחב שונה, $\Re^{16}$, $\Re^8$, ו-$\Re^4$, בהתאמה.

20	12	4	12

(ג)

22	18	10	14	6	2	14	10

(ב)

23	21	17	19	11	9	15	13	5	7	3	1	15	13	9	11

(א)

איור 2.3: פיענות עם (א) $B = 4$, (ב) $B = 2$, (ג) $B = 1$.

Fig. 2.3: Decoding with (a) $B = 4$, (b) $B = 2$, and (c) $B = 1$.

הקשר המדויק בין נקודות השבת השונות, והמובן שלו, הוא הנושא העיקרי של עבודה זו, והוא מוצג ומוסבר בהמשך.

פרק 3

ייצוג היררכי של האות

בפרק הקודם ראינו כי הקוד המתאר אות מסוים מכיל בצורה מפורשת את גודל בלוק-הטווח המתאים, דהיינו את B. ראינו גם כי פיענוח הקוד עם בחירה שונה של גודל בלוק-טווח, B, מוביל לטרנספורמציה שונה, עם נקודת שבת שונה. בפרק זה נתאר תחילה קשר מפורש בין נקודות השבת השונות, הנובעות מבחירות שונות של B. לאחר מכן, נראה שקיימת פונקציה (של משתנה רציף!), אשר נובעת חד-ערכית מתוך הקוד. פונקציה זו בסיסית להבנת הייצוג ההיררכי, והיא מהווה מרכיב עיקרי בעבודה זו.

3.1 פיענוח קוד ברזולוציות שונות

המשפט הבא מתאר את הקשר בין שתי נקודות שבת שונות של קוד, כאשר משתמשים בגודל בלוק-טווח שונה.

משפט 2 (**קשר היררכי**) יהי נתון קוד IFS, אשר מוביל ל-W^1 עם $B = B_1$, ומוביל ל-$W^{\frac{1}{2}}$ עם $B = B_1/2$. נקודות השבת של הטרנספורמציות האלה יסומנו כ-$\vec{f}^1$ ו-$\vec{f}^{\frac{1}{2}}$, בהתאמה. אזי:

ירידה ברזולוציה (Zoom-out) :

$$\vec{f}^{\frac{1}{2}}(j) = \frac{1}{2}\left\{\vec{f}^{1}(2j) + \vec{f}^{1}(2j-1)\right\}, \qquad j = (1, \cdots, \frac{N_1}{2}) \tag{3.1}$$

כאשר $N_1 \triangleq M_R \cdot B_1$.

עלייה ברזולוציה (Zoom-in) :

$$\vec{f}^{1}((i-1)B_1 + j) = a_i \vec{f}^{\frac{1}{2}}((m_i - 1){D_h}^{\frac{1}{2}} + j) + b_i \tag{3.2}$$

$$i = (1, \cdots, M_R), \qquad j = (1, \cdots, B_1)$$

כאשר ${D_h}^{\frac{1}{2}} \triangleq \frac{{D_h}^1}{2} = \frac{B_1}{2}$.

למשפט זה ניתנות שתי הוכחות בנספחים: הוכחה ישירה, שהיא גם יותר כללית, ניתנת בנספח (א). הוכחה נוספת, יותר פשוטה, ניתנת בנספח (ג), אולם דורשת משפט שמוכח רק בהמשך (משפט הפונקציה המוכלת בקוד).

הסבר למשפט: כדי לחשב כל איבר של $\vec{f}^{\frac{1}{2}}$, כאשר $\vec{f}^{1}$ נתון, יש לקחת את הממוצע של שני איברים סמוכים ב-$\vec{f}^{1}$, כפי שמתואר ב-3.1. מצד שני, כדי לחשב איברים של $\vec{f}^{1}$ כאשר $\vec{f}^{\frac{1}{2}}$ ידוע, יש לבצע את משוואה (3.2) , שדומה מאד להפעלת W^1 עצמו (השווה למשוואה (4.24)).

משפט 2 מתאר קשר בין זוג האותות $\vec{f}^{1}$ ו-$\vec{f}^{\frac{1}{2}}$. אותו קשר קיים גם בין זוג האותות $\vec{f}^{\frac{1}{2}}$ ו-$\vec{f}^{\frac{1}{4}}$, כלומר

$$\vec{f}^{\frac{1}{4}}(j) = \frac{1}{2}\left\{\vec{f}^{\frac{1}{2}}(2j) + \vec{f}^{\frac{1}{2}}(2j-1)\right\}, \tag{3.3}$$

$$\vec{f}^{\frac{1}{2}}((i-1)B_{\frac{1}{2}}+j)=a_i\vec{f}^{\frac{1}{4}}((m_i-1){D_h}^{\frac{1}{4}}+j)+b_i. \qquad (3.4)$$

אותו קשר גם מתקיים עבור הזוג $\vec{f}^{\frac{1}{4}}$ ו-$\vec{f}^{\frac{1}{8}}$, וכך הלאה. אוסף נקודות השבת מתאר אם כך מבנה היררכי, כפי שתואר באיור 2.3. מבנה זה נקרא פירמידת נקודות השבת של ה-IFS, כאשר $\vec{f}^{1/2^p}$ מרכיב את הרמה ה-p-ית של הפירמידה, כך שב-$\vec{f}^{1/2^p}$ יש $N/2^p$ איברים. כלאמר, האות המקורי באורך N מתאים לרמה $p=0$ בפירמידה. הרמה בעלת הרזולוציה הנמוכה ביותר, איור 2.3 (c), נקראת רמה עליונה (*top-level*).

הקשר בין שתי רמות סמוכות בפירמידה, ניתן לסיכום באופן הבא:

1. כדי לעלות בפירמידה, מרמה p לרמה $p+1$, יש לבצע את הפעולה הבאה (Zoom-out):

$$\vec{f}^{1/2^{p+1}}(j)=\frac{1}{2}\left\{\vec{f}^{1/2^p}(2j)+\vec{f}^{1/2^p}(2j-1)\right\}, \qquad (3.5)$$

$$j=(1,\cdots,N_1/2^{p+1}).$$

(והתוצאה מתלכדת עם משוואה (3.1) עבור $p=0$).

2. כדי לרדת בפירמידה, מרמה $p+1$ לרמה p, יש לבצע את הפעולה הבאה (Zoom-in):

$$\vec{f}^{1/2^p}((i-1)B_{1/2^p}+j)=a_i\vec{f}^{1/2^{p+1}}((m_i-1){D_h}^{1/2^{p+1}}+j)+b_i \qquad (3.6)$$

$$i=(1,\cdots,M_R), \qquad j=(1,\cdots,B_{1/2^p}).$$

כאשר $B_1/2^{p+1}={D_h}^1/2^{p+1}\triangleq{D_h}^{1/2^{p+1}}$. (משוואה (3.2) מתאימה למקרה $p=0$).

נקודת מבט שונה על הנוסחאות מושגת ע"י כך שנשים לב שבלוקי-התחום של $\vec{f}^1$, אחרי כיווץ מרחבי, הם למעשה הבלוקים המרכיבים את $\vec{f}^{\frac{1}{2}}$. באופן פורמלי, ניתן להראות שאכן הדבר כך ע"י כתיבה מפורשת של האיבר ה-i בבלוק-תחום ה-m_i של $\vec{f}^1$, לאחר כיווץ

מרחבי (2.15) - (2.16):

$$\varphi(\vec{d}^{1}_{m_i})(l) = \frac{1}{2} \cdot \left\{ \vec{f}^{1}((m_i - 1){D_h}^{1} + 2l - 1) + \vec{f}^{1}((m_i - 1){D_h}^{1} + 2l) \right\}, \tag{3.7}$$

$l \in (1, \cdots, B_1)$.

לפי משוואה (3.1) , הצד הימני של המשוואה האחרונה הוא בדיוק $\vec{f}^{\frac{1}{2}}((m_i - 1){D_h}^{1}/2 + l)$, ואם גם נסמן, כמקודם, ${D_h}^{\frac{1}{2}} \triangleq {D_h}^{1}/2$, ונסמן את $\vec{d}^{\frac{1}{2}}_{m_i}$ כבלוק-תחום ה-m_i-י של $\vec{f}^{\frac{1}{2}}$, אנו מסיקים:

$$\varphi(\vec{d}^{1}_{m_i})(l) = \vec{d}^{\frac{1}{2}}_{m_i}(l), \qquad l \in (1, \cdots, B_1). \tag{3.8}$$

מתעוררת השאלה, מהו הגודל הכי קטן של הרמה-העליונה (top-level) כך שהקשרים הנ"ל בין שתי רמות שכנות עדיין יתקימו ? על שאלה זו עונה ההיקש הבא.

היקש 1 תהי $\vec{f}^{1}$ נקודת שבת ב-$\Re^{N}$ של קוד IFS נתון כלשהו, ויהי $B = B_1 = 2^{l}$, $D = D_1 = 2B$ וגם $D_h = B_1$, אזי מספר הרמות בפירמידת נקודות השבת של קוד ה-IFS הוא

$$log_2(B) + 1 = l + 1 \tag{3.9}$$

ומספר זה מוביל לרמה עליונה באורך $(= M_R) \quad N/2^{l}$.

הוכחה : עליה ברמה אחת בפירמידה, משמעותה הקטנת גודל בלוק הטווח, B, פי 2. מכיוון שגודל זה חייב להיות לפחות 1 כדי שניתן יהיה לעבוד עם ה-IFS, ההיקש נובע מידית. מ.ש.ל. ■

טבלה 3.1 מסכמת את הביטויים וההגדרות שנתקבלו והוגדרו עד כה.

נקודת-שבת $\vec{f}$	מס. איברים N	גודל בלוק-טווח B	רמה מספר. p
$\vec{f}^{1}$	$N_1 = M_R \cdot B_1$	B_1	0
$\vec{f}^{1/2}$	$N_{1/2} = N_1/2$	$B_{1/2} = B_1/2$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\vec{f}^{1/B_1}$	$N_{1/B_1} = M_R$	$B_{1/B_1} = 1$	$log_2 B_1 = l$ (רמה-עליונה)

טבלה 3.1: פירמידת נקודות-שבת: הגדרות וסימונים.

Table 3.1: Pyramid of fixed-points: Notation summary.

3.2 הפונקציה המוכלת בקוד (IFS embedded function)

הצגת האות ברזולוציות שונות קשורה ישירות לדגימת פונקציה של משתנה רציף, והצגתה ע"י מספר סופי של ערכים. בהגדרה הבאה, מובאת שיטת דגימה המבצעת מיצוע על ערך האות בתחום נתון (שיטה זו היא מודל מקובל לדגימת תמונה [32]).

הגדרה 1 תהי נתונה פונקציה $G(x) \in L^\infty[0,1]$, נגדיר את $G_r(i)$ באופן הבא:

$$G_r(i) \triangleq r \int_{(i-1)\frac{1}{r}}^{i\frac{1}{r}} G(x)\,dx, \qquad i = 1, \cdots, r. \tag{3.10}$$

$G_r(i)$ נקראת הפונקציה $G(x)$ ברזולוציה r. אנו אומרים ש-$G_{r_1}(i)$ היא עדינה יותר (כלומר, עם רזולוציה גבוהה יותר,(finer)) מאשר $G_{r_2}(i)$ שהיא גסה יותר (coarser) אם $r_1 > r_2$.

המשפט הבא מציג את המושג החדש של הפונקציה המוכלת בקוד, וקושר אותה לפירמידה של נקודות-שבת.

משפט 3 (**הפונקציה המוכלת בקוד**) בהינתן קוד IFS, קיימת פונקציה אחת ויחידה $G(x) \in L^\infty[0,1]$, כך שהוקטור $\vec{f}_N \in \Re^N$ הוא נקודת שבת של הקוד אם ורק-אם הוא שווה לפונקציה $G(x)$ ברזולוציה $r = N$ לכל N, כלומר,

$$\vec{f}_N(j) = G_N(j), \qquad j = 1, 2, \cdots, N \quad . \tag{3.11}$$

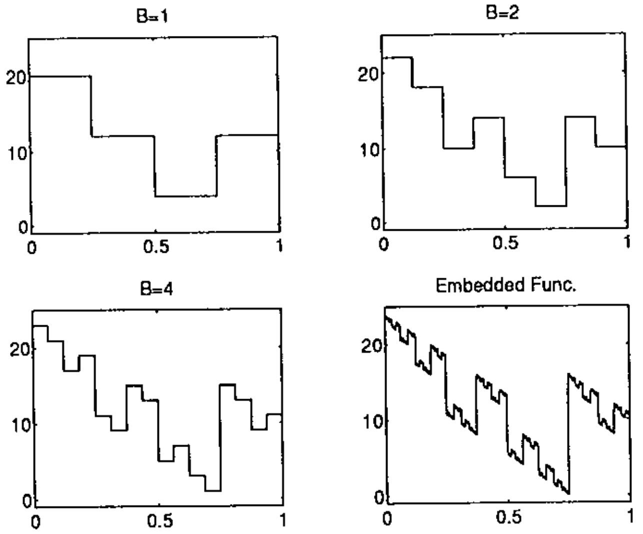
הפונקציה $G(x)$ נקראת <u>**הפונקציה המוכלת בקוד**</u> (<u>IFS embedded function</u>).

המשפט מוכח בנספח (ב).

איור 3.1 מדגים את מושג הפונקציה המוכלת בקוד. הקוד הוא אותו קוד שתואר מקודם, ראה איור 2.1. נקודות השבת של הקוד, עבור $B = 1$, $B = 2$, ו-$B = 4$, מודגמות באיור 3.1 כפונקציה של משתנה רציף $x \in [0,1]$. לדוגמא, נקודת השבת עבור $B = 1$ היא הוקטור $[20, 12, 4, 12]$, בעל $N = 4$ איברים. וקטור זה מתואר כפונקציה קבועה למקוטעין:

$$f(x) = \begin{cases} 20 & x \in [0, \frac{1}{4}) \\ 12 & x \in [\frac{1}{4}, \frac{1}{2}) \\ \vdots & \vdots \end{cases} \tag{3.12}$$

הפונקציה המוכלת בקוד $G(x)$ מתוארת אף היא. מתוך האיור ניתן לראות שנקודות השבת 'שואפות' לפונקציה המוכלת בקוד. יתרה מזו, ניתן להבחין שהערך של הפונקציות הקבועות למקוטעין, המתארות נקודות-שבת, שווה לממוצע של הפונקציה המוכלת, על פני קטעים מתאימים, כפי שמתואר במשוואות (3.10) - (3.11) .



איור 3.1: נקודות שבת עבור $B = 1, B = 2, B = 4$, והפונקציה המוכלת בקוד.

Fig. 3.1: Fixed-points for $B = 1, B = 2, B = 4$,and the corresponding IFS embedded Function.

3.2.1 קשר לפונקצית אינטרפולציה פרקטלית

פונקצית אינטרפולציה פרקטלית (פא"פ) (Fractal-Interpolation Function), המתוארת ב-[4, 6], מזכירה את מושג הפונקציה המוכלת בקוד. הפא"פ היא נקודת שבת של קוד IFS, אשר מתואר באופן דומה לקוד המתאר את הפונקציה המוכלת בקוד, דהיינו מיפוי של אזורי-תחום אל אזורי-טווח (אזור, ולא בלוק, משום שהמדובר בפונקציה של משתנה רציף). הקוד נבנה כך שהפא"פ הנוצרת תעבור דרך מספר נקודות הנתונות מראש, אלו הן הנקודות שהפונקציה מבצעת אינטרפולציה שלהן. כאמור, מבנה הקוד המתאר את הפא"פ דומה לזה של הפונקציה המוכלת בקוד, אולם להלן נציין מספר הבדלים עקרוניים ביניהן (קוד פא"פ שאליו נתייחס הוא מתוך [4] עמ' 219 – 218, והקוד המתאר פונקציה מוכלת בקוד הוא

בנספח B נוסחאות B.4 ו-B.5):

- הפא"פ היא פונקציה רציפה (בהכרח!), ואילו הפונקציה המוכלת בקוד אינה בהכרח רציפה (לרוב לא). הסיבה לכך נובעת מאופן מציאת הקוד עבור הפא"פ: הקוד נבנה כך שערך הפונקציה בנקודת אינטרפולציה יתלכד עם הערך הנתון, וכיוון שכל נקודה שייכת למעשה לשני בלוקי-טווח סמוכים, הרי שהפונקציה רציפה במעבר בין שני בלוקי-טווח סמוכים. כיוון שאי-הרציפות המקסימלית תתרחש במעבר בין שני בלוקי-טווח סמוכים, הרי שהפונקציה רציפה. השיוויון בנקודות האינטרפולציה אינו נכון לגבי בניית הקוד של הפונקציה המוכלת בקוד, אשר אינו קשור לערכים בנקודות מסוימות (למעשה, הראינו שהוא קשור לאינטגרל הפונקציה על-פני תחום מתאים). ראה גם הערה בהמשך לגבי המקרה של דגימה-נקודתית.

כלומר, מבנה הקוד בשני המקרים הוא אותו דבר (למעט גבולות האזורים, כפי שנדון מיד), אולם הפרמטרים, שהם אלה שקובעים האם הפונקציה רציפה או לא, נמצאו בפא"פ כך שהפונקציה תהיה רציפה, ואילו במקרה של הפונקציה המוכלת בקוד, לא הוצב אילוץ זה. עובדה זו באה לידי ביטוי בכך שאזור הטווח מוגדר כתחום סגור בפא"פ, כלומר מכיל את שתי נקודות הקצה, בעוד בפונקציה המוכלת בקוד אזורי הטווח מוגדרים כתחום חצי-פתוח, כך שהם מכילים נקודת קצה אחת בלבד.

- ב-[4], עמ' 223 – 224, מופיעה נוסחה לחישוב ישיר של ערך האינטגרל של פא"פ עבור תחום מסוים. נוסחה זו מזכירה את העובדה שהראינו, ולפיה ניתן לחשב את ערך אינטגרל הפונקציה בעזרת מציאת נקודת השבת המתאימה. ביתר פרוט, יש למצוא רזולוציה מתאימה N, כך שהתחום עבורו רוצים לבצע אינטגרציה $[a, b]$, יתלכד עם נקודות השריג $[P/N, Q/N]$, ואז ערך האינטגרל הוא ממוצע של $(Q - P)$ האיברים

המתאימים בנקודת השבת. ההבדל העקרוני הוא בעובדה שבמקרה של פא"פ מופיעה נוסחה סגורה לחישוב הערך, ללא צורך במציאת נקודת שבת דיסקרטית. הסיבה לכך היא שהפא"פ המתוארת שם עוסקת במיפוי כל התחום של הפונקציה אל כל בלוק-טווח, כלומר הפא"פ מתארת מקרה בו יש בלוק-תחום אחד, והוא כל התחום $[0,1]$.

הערה נוספת, שתובהר רק בהמשך, היא העובדה שהפא"פ ונקודות האינטרפולציה, עבור המקרה של שריג נקודות-אינטרפולציה אחיד ושימוש בטרנספורמציות ליניאריות, מתלכדות עם הפונקציה המוכלת בקוד ונקודת השבת הדיסקרטית, עבור שימוש בדגימה נקודתית וב-$B = 1$. לכן, המשפטים המתוארים כאן מציעים דרך לחשב את ערכי פונקצית האינטרפולציה בנקודות שונות, בעזרת חישוב נקודת שבת דיסקרטית, ללא צורך בחישוב הפונקציה כולה.

פרק 4

נסוח מטריצי של הקוד והרחבות לשיטת הקידוד

4.1 נסוח מטריצי

בחלק זה יוצג ניסות מטריצי של קוד IFS, כלומר של הטרנספורמציה W. התאור המטריצי, כפי שהוא מופיע פה, משמש לניסות משפט המתאר חסם על המימד הפרקטלי של הפונקציה המוכלת בקוד, ולחישוב מקדם הכיווץ (s) של הטרנספורמציה, כפי שנראה בהמשך. כמו כן הייצוג ישים למספר שימושים המוזכרים בקצרה בפרק הדן בכיווני המשך לעבודה.

יהי נתון קוד IFS הכולל את כל המידע הדרוש לחישוב W, כפי שתואר בסעיף 2.2.1.

הטרנספורמציה $\vec{u} = W(\vec{v})$ מורכבת מהאוסף $\{w_i\}_{i=1}^{M_R}$, כאשר כל w_i פועל על בלוק-תחום מתאים. האיבר ה-j-י בבלוק-טווח מספר i, השייך לוקטור $\vec{u}$, נתון לפי (2.13):

$$\vec{r}_i(j) = \vec{u}((i-1)\cdot B + j) \quad ; \quad i = (1,\cdots,M_R), \qquad j = (1,\cdots,B) \qquad (4.1)$$.

איבר זה, לפי (4.24), הוא תוצאה של טרנספורמציה על בלוק-התחום המתאים, כפי שנתון

בקוד:

$$.\vec{r}_i(j) = a_i \cdot \varphi(\vec{d}_{m_i})(j) + b_i \tag{4.2}$$

נחליף את φ בהגדרתה לפי (2.16) , ונשתמש במשוואה (2.15) :

$$.\vec{r}_i(j) = a_i \cdot \frac{1}{2} \cdot (\vec{v}((m_i - 1)D_h + 2j) + \vec{v}((m_i - 1)D_h + 2j + 1)) + b_i \tag{4.3}$$

את המשוואה האחרונה נכתוב כעת בצורה מטריצית, עבור כל איברי $\vec{u}$, ונקבל:

$$,\vec{u} = [F]\vec{v} + \vec{b} \tag{4.4}$$

כאשר המטריצה $[F]$ והוקטור $\vec{b}$ מתוארים להלן:

- $\vec{b}_{N\times 1}$ - וקטור הסחה. זהו וקטור המורכב מ-$(M_R = N/B)$ בלוקים בסיסיים:

$$\vec{b} = \begin{bmatrix} \vec{b}_1^T & \vec{b}_2^T & \cdots & \vec{b}_i^T & \cdots & \vec{b}_{N/B}^T \end{bmatrix}^T \tag{4.5}$$

כאשר,

$$.\vec{b}_i = b_i \cdot \underbrace{[\begin{array}{cccccc} 1 & 1 & 1 & \cdots & 1 & 1 \end{array}]}_{B\ times}^T, \qquad i = (1, 2, \cdots, \frac{N}{B}) \tag{4.6}$$

- $[F]_{N\times N}$ - מטריצת העברה, הניתנת לכתיבה:

$$.[F] = [A] \cdot \frac{1}{2} \cdot [D] \tag{4.7}$$

$[D]$ - מטריצת מיפוי בלוקי-תחום. המכפלה $(\frac{1}{2}[D])\cdot\vec{v}$ ממלאת את תפקיד הפונקציה המכווצת $\varphi(\vec{d}_{m_i})$, ולכן מטפלת בשני היבטים:

1. הפקת $\vec{d}_{m_i}$ מתוך $\vec{v}$.
2. ביצוע כיווץ-מרחבי על $\vec{d}_{m_i}$.

$[D]$ היא מטריצה המורכבת ממטריצות (block-matrix), ונתאר אותה בעזרת דוגמא. בדוגמא זו, בלוק-תחום $\vec{d}_3$ ממופה לבלוק-טווח $\vec{r}_1$, $\vec{d}_1$ ממופה ל-$\vec{r}_2$, ובלוק $\vec{d}_4$ ממופה ל-$\vec{r}_3$:

$$[D] = \begin{bmatrix} [0]_B & [0]_B & \boxed{[D]_{B\times D}} & [0]_B & [0]_B & [0]_B & [0]_B & \cdots \\ \boxed{[D]_{B\times D}} & [0]_B & [0]_B & [0]_B & [0]_B & [0]_B & [0]_B & \cdots \\ [0]_B & [0]_B & [0]_B & \boxed{[D]_{B\times D}} & [0]_B & [0]_B & [0]_B & \cdots \\ \cdots & & \cdots & & \cdots & & \cdots & \ddots \end{bmatrix}_{N\times N} \tag{4.8}$$

המטריצה $[0]_B$ היא מטריצת אפסים בגודל $B \times B$. מיקום המטריצות $[D]_{B\times D}$ (המרכיבות את $[D]$), נקבע ע"י הפרמטרים m_i, והכיווץ המרחבי מבוצע ע"י איברי המטריצה, כפי שמתואר להלן:

1. אם בלוק-תחום m_i ממופה לבלוק-טווח i, אזי נמקם את המטריצה $[D]_{B\times D}$ כך שהאיבר השמאלי-עליון שלה ימוקם באיבר

$$((i-1)B+1,(m_i-1)D_h+1) \tag{4.9}$$

של $[D]$.

2. פונקצית הכיווץ המרחבי מבוצעת ע"י $[D]_{B\times D}$, בעלת המבנה הבא:

$$[D]_{B\times D}=\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & \cdots \\ \vdots & & & & & & \ddots & \vdots \\ 0 & 0 & & & & 0 & 1 & 1 \end{bmatrix}_{B\times D} \qquad (4.10)$$

כלומר, $\frac{1}{2}[D]_{B\times D}$ ממפה בלוקים בגודל $D=2B$ לבלוקים בגודל B ע"י מיצוע של איברים סמוכים.

$[A]$ - מטריצת ביול. זוהי מטריצה אלכסונית (diagonal matrix), המורכבת ממטריצות אלכסוניות (block-diagonal matrix):

$$[A]=\begin{bmatrix} [A_1] & [0]_B & [0]_B & \cdots \\ [0]_B & [A_2] & [0]_B & \cdots \\ [0]_B & [0]_B & [A_3] & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}_{N\times N} \qquad (4.11)$$

כאשר:

$$,[A_i]=a_i\cdot[I]_{B\times B}, \qquad i=(1,2,\cdots,\frac{N}{B}) \qquad (4.12)$$

$[I]_{B\times B}$ היא מטריצת היחידה ממימד $B\times B$, והמטריצה $[0]_B$ היא מטריצת אפסים מאותו מימד. אם, עבור קוד מסוים, $\forall i \quad a_i=a$, אזי $[A]=a\cdot[I]_{N\times N}$.

מימדי המטריצות $[F],[A],[D]$, כמו גם מימד הוקטור $\vec{b}$, תלויים ישירות בגודל B. ואכן, כפי שכבר ראינו, שינוי ערכו של B מוביל לטרנספורמציה W שונה, ולכן גם למטריצות

שונות, אם כי המבנה הבסיסי שלהן נשאר אותו דבר. לדוגמא, איור 4.1 מתאר את מרכיבי הניסוח המטריצי של קוד ה-IFS שתואר באיור 2.1, עם $B = 2$.

$$[A] = \tfrac{1}{2}\cdot[I]_{8\times 8}, \qquad \vec{b} = [12,12,8,8,0,0,4,4,]^T$$

$$[D] = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \; ; \qquad \begin{array}{ccc} \vec{d}_1 & \rightarrow & \vec{r}_1 \\ \vec{d}_3 & \rightarrow & \vec{r}_2 \\ \vec{d}_2 & \rightarrow & \vec{r}_3 \\ \vec{d}_1 & \rightarrow & \vec{r}_4 \end{array}$$

איור 4.1: ייצוג מטריצי של הקוד - המטריצות $[A]$,$[D]$, והוקטור $\vec{b}$, המתארים את הקוד המתואר באיור 2.1, עבור $B = 2$.

Fig. 4.1: Matrix representation of the code - The matrices $[A]$ and $[D]$, and the vector $\vec{b}$ describing the IFS-code of Fig. 2.1, for $B = 2$.

4.1.1 מציאת מקדם כיווץ

נביא כעת דוגמא לשימוש בייצוג המטריצי למציאת מקדם-הכיווץ של קוד ה-IFS.

נעסוק במרחב הנורמי $(\Re^n, \|\cdot\|)$, כאשר הנורמה משרה את המרחק המתאים על המרחב (למשל, $\|\cdot\|_2$ משרה את המרחק $d_2(\cdot,\cdot)$). יהי נתון הקוד W, בעל ייצוג מטריצי כמתואר

מקודם, דהיינו

$$.\vec{u} = [F]\vec{v} + \vec{b}$$

גורם הכיווץ של הטרנספורמציה, s, שאותו נרצה לבטא בעזרת הייצוג המטריצי, הוא המספר המינימלי המקיים

$$.\forall\, \vec{x},\vec{y} \in \Re^n \quad d(W(\vec{x}), W(\vec{y})) \leq s\, d(\vec{x}, \vec{y}) \tag{4.13}$$

לצורך זה, נציג את W מפורשות,

$$\begin{aligned} d(W(\vec{x}), W(\vec{y})) &= \|W(\vec{x}) - W(\vec{y})\| \\ &= \|[F]\vec{x} + \vec{b} - [F]\vec{y} - \vec{b}\| \\ &= \|[F](\vec{x} - \vec{y})\| \\ &\leq \|[F]\| \cdot \|\vec{x} - \vec{y}\| \end{aligned}$$

כאשר האי-שיוויון האחרון נובע מהגדרת הנורמה של אופרטור (מטריצה במקרה הזה). ומתקבל לכן,

$$.s = \|[F]\| \tag{4.14}$$

לצורך ביטוי מפורש אף-יותר, נסמן ב-f_{ij} את האיבר ה-(i, j) של המטריצה $[F]$, ונניח כי המטריצה מגודל $N \times N$, ומתקבל [72] :

$$\|[F]\|_1 = \max_{1 \leq j \leq N} \sum_{i=1}^{N} |f_{ij}| \tag{4.15}$$

$$\|[F]\|_\infty = \max_{1 \leq i \leq N} \sum_{j=1}^{N} |f_{ij}| \tag{4.16}$$

$$\|[F]\|_2 = \sqrt{|\lambda_{\max}|} \tag{4.17}$$

כאשר בביטוי האחרון $\lambda_{\max}$ הוא הערך העצמי הגדול ביותר של $[F]^T \cdot [F]$.

4.2 הרחבות לשיטת הקידוד

בחלקו הראשון של סעיף זה נתאר בקצרה עבודות שונות שבוצעו בעקבות הרעיון הבסיסי של Jacquin. בחלק השני נתאר ביתר פרוט שיטה מסוימת, שרלבנטית יותר להמשך.

מאמר סקירה על הנושא הוא [31].

4.2.1 הרחבות כלליות

כאמור, נסקור תחילה בקצרה מספר שיפורים\הרחבות.

- מציאת מקדמי הטרנספורמציה (a_i, b_i). תהליך מציאת המקדמים מורכב ממינימיזציה של המרחק בין בלוק-טווח לבלוק-תחום אחרי הטרנספורמציה. אולם, עדיין נותרת הבעיה של כימות ערכי הפרמטרים (הקצאת סיביות). Fisher at al. , ב-[21], מצאו כי מספר מתאים של סיביות הוא 5 עבור b_i, ו-7 עבור a_i. כמו כן, באותו מקום, דנים גם באפשרות להשתמש במקדמי כיול a_i גדולים מ-1. ניתן לעשות זאת, כל עוד הטרנספורמציה כולה נשארת מכווצת לבסוף (eventually contractive). הערך המקסימלי שנמצא שם בעזרת סימולציות, עבור ערכי a_i, הוא 1.5.
- חלוקת התמונה לבלוקים - בנושא זה מתייחסים לחלוקת התמונה לבלוקי-טווח בגדלים שונים. דרך בסיסית היא לבצע חלוקת Quadtree [21, 29]. למשל, מתחילים בחלוקה וקידוד של התמונה עם גודל בלוק של $B = 16$. אם קיים בלוק-טווח שעבורו לא נמצאה התאמה מספיק טובה (לפי קריטריון מסוים), אזי בלוק זה יחולק שוב, ויבוצע מחדש תהליך מציאת התאמה עבור בלוק זה, כאשר הפעם גודל הבלוק הוא

חצי מהקודם.

נזכור כמובן שבלוקים גדולים ייתרמו לדחיסה גבוהה, אולם התאמתם תהיה פחות טובה, ולכן האיכות פחות טובה. באופן כללי, שיטה זו הביאה לשיפורים בביצועי דחיסה, אולם האפקט של בלוקים גדולים בתמונה המשוחזרת מפריע מאד.

דרך נוספת היא חלוקה לצורות שונות (למשל, משולשים במקום מרובעים, או מלבנים במקום מרובעים). התוצאות של נסיונות אלה הראו שיפור באיכות התמונה (פחות הפרעות של blockiness) [21].

- זמן קידוד - בעזרת מיון בלוקי התחום (ה-Domain-pool), למשל לבלוקים המתארים אזור חלק, מרקם ושפה, ניתן להקטין את זמן הקידוד. עבור בלוק-טווח מסוים, נקבע מאיזה סוג הוא, ונחפש לו התאמה רק בקבוצת בלוקי-התחום הדומים לו [36]. שיטה זו מאיצה מאד את זמן הקידוד, וגורמת לירידה קלה בלבד באיכות הדחיסה.

  שיטה נוספת לקיצור זמן החיפוש היא קביעת סף שגיאה מסוים, כך שברגע שהשגיאה בין בלוק-התחום והבלוק-טווח קטנה מערך הסף, מופסק החיפוש עבור בלוק טווח זה [21].

4.2.2 אורתוגונליזציה ביחס לבלוק קבוע

בתת-סעיף זה נתאר הרחבה נוספת לאלגוריתם הבסיסי, הרחבה שהוצעה ע"י Øien [50]. הרחבה זו עוסקת באורתוגונליזציה של בלוקי התחום (domain-blocks) ביחס לבלוקים אבי-טיפוס. לצורכנו אנו נסתפק באורתוגונליזציה יחסית לבלוק אב-טיפוס קבוע אחד (בלוק D.C.).

ההבחנה שעומדת בבסיס ההרחבה היא שתהליך הקידוד מורכב מנסיון לקרב וקטור

(בלוק-טווח) בעזרת וקטור אחר (בלוק-תחום) ופרמטרים שונים, או כפי שרשמנו 4.24

$$\vec{r}_i = a_i\varphi(\vec{d}_{m_i}) + b_i\vec{1}_B.$$

לצורך מציאת המקדמים a_i ו-b_i יש לפתור בעית מינימיזציה.

לעומת זאת, נכתוב כעת את המשוואה האחרונה בצורה שונה,

$$\vec{r}_i = a_i\vec{v}_1 + b_i\vec{v}_2$$

כאשר

$$\vec{v}_1 = \varphi(\vec{d}_{m_i}), \qquad \vec{v}_2 = [\underbrace{1\ \ 1\ \ 1\ \ \cdots\ \ 1\ \ 1}_{B\ times}]^T$$

כדי להקל על מציאת המקדמים a_i ו-b_i, נגדיר וקטור חדש $\vec{v}_3$, אשר הוא התוצאה של אורתוגונליזציה של $\vec{v}_1$ יחסית ל-$\vec{v}_2$, כלומר $\vec{v}_3 \perp \vec{v}_2$, ואז

$$\vec{r}_i = \tilde{a}_i\vec{v}_3 + \tilde{b}_i\vec{v}_2.$$

כעת, מציאת המקדמים היא יותר פשוטה, מכיוון שיש רק להטיל את $\vec{r}_i$ על כל אחד מהכיוונים הבלתי-תלויים (עבור מציאת מינימום ב- l^2)!

נשים לב שבמקרה המסוים אותו אנו מתארים, כאשר $\vec{v}_2$ הוא קבוע ו-$\vec{v}_1 = \varphi(\vec{d}_{m_i})$, הרי שמשמעות האורתוגונליזציה היא הסרת רכיב ה-D.C. מוקטור בלוק-התחום.

כלומר, תהליך הקידוד והפיענוח יהיה דומה לתהליך הרגיל, אלא שהפעם, לפני כל שמוש בבלוק-תחום, יש להסיר ממנו את רכיב ה-D.C..

נתאר כעת במפורט את תהליכי הקידוד והפיענוח בשיטה החדשה. הסיכום הבא, המתאר את תהליך הקידוד, מבוסס על סיכום-4 (ראה עמוד 22), עם ההתאמות הדרושות.

סיכום 5 : קידוד IFS-עם-אורתוגונליזציה של $\vec{\mu}_o$

Summary 5 : IFS-with-orthogonalization Encoding of $\vec{\mu}_o$

1. שמור B בקובץ הקוד.

2. שמור M_R בקובץ הקוד, כאשר $M_R = N/B$, ו-N הוא האורך של $\vec{\mu}_o$.

3. חלק את $\vec{\mu}_o$ ל-M_R בלוקי-טווח,

$$\vec{r}_i(j) = \vec{\mu}_o((i-1)\cdot B + j), \qquad (4.18)$$

$$i = (1, \cdots, M_R), \qquad j = (1, \cdots, B).$$

4. הפק מתוך $\vec{\mu}_o$ את $M_D = (\frac{N-D}{D_h} + 1)$ בלוקי-התחום,

$$\vec{d}_l(j) = \vec{\mu}_o((l-1)D_h + j), \qquad (4.19)$$

$$l = 1, 2, \cdots, M_D, \qquad j = 1, 2, \cdots, D.$$

5. הסר מתוך M_D בלוקי-התחום את רכיב ה- D.C. שלהם, כלומר

$$,\vec{d}_l^{ort}(j) = \vec{d}_l(j) - \frac{1}{D}\sum_{j=1}^{D} \vec{d}_l(j) \qquad (4.20)$$

$$l = 1, 2, \cdots, M_D.$$

6. עבור $i = 1, \ldots, M_R$:

   (א) מצא את הפרמטר b_i לפי

$$b_i = (\text{D.C. value of range block } i) = \frac{1}{B}\sum_{j=1}^{B} \vec{r}_i(j) \qquad (4.21)$$

(ב) מצא את הפרמטרים האופטימליים (a_i, m_i), כלומר את אלה שיביאו למינימום את

$$.d^{\infty}(\vec{r}_i, (a_i\varphi(\vec{d}^{ort}_{m_i}) + b_i\vec{1}_B)) \qquad (4.22)$$

(ג) שמור את הפרמטרים (a_i, b_i, m_i) בקובץ הקוד.

תהליך הפיענוח בעזרת איטרציות דומה לתהליך הפיענוח הרגיל. לתאור מפורט נחזור על סיכום-1 (ראה עמוד 18), המתאר ביצוע איטרציה בודדת, עם השינויים הדרושים.

סיכום 6 : אלגוריתם לביצוע הטרנספורמציה $\vec{u} = W^{ort}(\vec{v})$

Summary 6 : Algorithm for Performing the Transformation $\vec{u} = W^{ort}(\vec{v})$

1. עבור $i = 1, \ldots, M_R$

(א) הפק את בלוק-התחום $\vec{d}_{m_i}$ מתוך הוקטור $\vec{v}$.

(ב) הסר מבלוק-התחום $\vec{d}_{m_i}$ את רכיב ה- D.C. שלו,

$$\vec{d}^{ort}_{m_i} = \vec{d}_{m_i} - (\text{D.C. value of } \vec{d}_{m_i}) \qquad (4.23)$$

(ג) חשב את בלוק-הטווח

$$.\vec{r}_i = w_i{}^{ort}(\vec{d}_{m_i}) = a_i\varphi(\vec{d}^{ort}_{m_i}) + b_i\vec{1}_B \qquad (4.24)$$

2. שרשר את בלוקי–הטווח, $\vec{r}_i$, $i = (1, \cdots, M_R)$, לפי סדר ה–i, כדי לקבל וקטור חדש, $\vec{u}$. אורך הוקטור $\vec{u}$ הוא : $N = M_R \cdot B$.

להלן נתאר מספר תכונות של שיטה זו:

- מציאת המקדמים בשיטה החדשה יותר פשוטה, עקב העובדה ששני הוקטורים, שבעזרתם מנסים לקרב את בלוק-הטווח, אורתוגונליים.
- המקדם b_i הוא ממוצע ערכי $\vec{r}_i$, כלומר רכיב ה-.D.C של בלוק-הטווח. לפיכך, חישוב מקדם זה אינו תלוי ב-$\vec{d}_{m_i}$, אלא רק ב-$\vec{r}_i$.
- יש לבצע הפחתת רכיב ה-.D.C מבלוקי-התחום גם בפיענוח.
- הרמה העליונה (Top-Level) בפירמידת נקודות השבת מתאימה לשימוש בגודל בלוק-טווח של $B = 1$. במקרה זה, כל בלוק טווח הינו איבר בודד, ואיבר זה הוא ממוצע ערכי הבלוק עבור ערך כלשהוא של B, כלומר רכיב ה-.D.C המתאים לבלוק. אולם, בשיטה החדשה, רכיב ה-.D.C של כל בלוק-טווח הוא למעשה המקדם b_i. לפיכך, את הרמה העליונה בפירמידה ניתן למצוא ישירות מתוך הקוד, ללא צורך בחישובים.

פרק 5

שימושים לייצוג ההירכי

מתוך הייצוג ההירכי של הקוד נובעים מספר יישומים שיתוארו בסעיפים הבאים. במקומות המתאימים תצורפנה דוגמאות.

5.1 פיענוח מהיר של הקוד

לשיטת הפיענות המהיר נקרא פיענוח הירכי, משום שהיא נובעת ישירות מתוך הייצוג ההירכי. בשיטת פיענות זו, מחשבים תחילה את הרמה העליונה (top-level) בפירמידה. חישוב הרמה יכול להעשות ע"י פיענות הקוד עם $B = 1$, כלומר, הפעלת הטרנספורמציה W על וקטור התחלתי באורך M_R, עד שמושגת נקודת השבת (או נקודה קרובה מספיק), או פיענות ע"י שמוש בערכי ה- D.C. של בלוקי הטווח b_i, אם עוקבים אחרי השיטה שתוארה בסעיף 4.2.2. אנו מניחים בהמשך פרק זה, אלא אם כן יצויין במפורש אחרת, שכל הפעולות (קידוד ופיענוח) מתבצעות ללא אורתוגונליזציה. לאחר חישוב הרמה העליונה, כל שיש לעשות כדי לעבור לרזולציה גבוהה יותר הוא לעקוב אחרי האלגוריתם המתואר במשוואה

(3.2) . חוזרים על תהליך המעבר לרזולוציה גבוהה יותר, עד אשר מושגת רמת הרזולוציה הדרושה (גודל הוקטור הדרוש). שיטה זו מושוות להלן לשיטה הרגילה של פיענוח בעזרת איטרציות (iterative decoding) [33], כאשר האיטרציות כולן מבוצעות על הוקטור (תמונה) ברזולוציה מלאה.

החיסכון במספר פעולות-חישוב בשיטת הפיענוח ההירכי נובע מכך שהאיטרציות מבוצעות רק ברמה העליונה, שהיא וקטור באורך קטן, בעוד שבשיטה הרגילה כל האיטרציות מתבצעות על וקטור באורך מלא.

5.1.1 חישוב מספר פעולות

לצורך ההמשך, נגדיר את הסימונים הבאים:

- S - זמן חישוב פעולת חיבור.
- M - זמן חישוב פעולת כפל.
- t_{op} - זמן חישוב כולל.
- I - מספר האיטרציות.

יתר על כן, הכפלות בחזקות של 2, כמו למשל $\frac{1}{2}$ או $\frac{1}{4}$, לא ייספרו כפעולות (ספירתן תראה ייתרון גדול עוד יותר לפיענוח ההירכי, משום שהכפלות אלה מופיעות בעיקר בחישוב $\varphi(\cdot)$, שאינו דרוש בשיטה ההירכית).

1. אות חד-מימדי (וקטור)

יהי N אורך הוקטור המקורי ויהי B גודל בלוק-הטווח בו.

- פיענוח בעזרת איטרציות

נתייחס למשוואות (4.24) ו- (2.16). עבור איטרציה אחת, זמן החישוב הוא:

$$.N \cdot [(1+1)S + 1M] = N \cdot [2S + 1M]$$

לכן, זמן החישוב הכולל הוא:

$$t^i_{op} = I \cdot N \cdot [2S + 1M] \qquad (5.1)$$

- פיענוח היררכי

לפי היקש 1 ברמה העליונה יש $\frac{N}{B} = M_R$ איברים. לכן, לפי התוצאה ב- (5.1), אנו יודעים את זמן החישוב הדרוש לחישוב הרמה העליונה. כמו כן, לפי משפט 1, ישנן $log_2(B)$ רמות בפירמידה, לא כולל את הרמה העליונה, כאשר ברמה ה-p יש $N/2^p$ איברים ($p = log_2(B)$ ברמה העליונה). לפי משוואה (3.2), ניתן כעת לחשב את עלות החישובים למעבר מהרמה העליונה אל וקטור באורך N:

$$.\sum_{p=0}^{log_2(B)-1} (\frac{N}{2^p}) \cdot [1S + 1M] = (B-1)\frac{N}{B}2 \cdot [S + M] \doteq N \cdot [2S + 2M] \qquad (5.2)$$

זמן החישוב הכולל הוא לכן:

$$.t^h_{op} \doteq N \cdot \left\{ \frac{I}{B}[2S + M] + [2S + 2M] \right\} \qquad (5.3)$$

2. <u>אות דו-מימדי (מערך,תמונה)</u>

יהי N^2 מספר האיברים בתמונה (מטריצה) המקורית, ויהי $B \times B$ גודל בלוק-הטווח בה.

- פיענוח בעזרת איטרציות

במקרה הדו-מימדי, ביצוע כיווץ מרחבי $\varphi(\cdot)$ דורש את חישוב הממוצע של 4- איברים.

עבור איטרציה אחת, זמן החישוב הוא:

$$.N^2 \cdot [(3+1)S + 1M] = N^2 \cdot [4S + 1M]$$

לכן, זמן החישוב הכולל הוא:

$$.t^i_{op} = I \cdot N^2 \cdot [4S + 1M] \tag{5.4}$$

- פיענוח הירכי

ברמה העליונה יש $(\frac{N}{B})^2$ איברים. לכן, לפי התוצאה (5.4), אנו יודעים את זמן החישוב הדרוש לחישוב הרמה העליונה. בדומה למקרה החד-מימדי, עלות המעבר מהרמה העליונה לרמה המכילה $N \times N$ איברים היא:

$$.\sum_{p=1}^{log_2(B)} (4^p(\frac{N}{B})^2)\cdot[1S + 1M] = \frac{4}{3}(B^2-1)(\frac{N}{B})^2\cdot[1S + 1M] \doteq N^2\frac{4}{3}\cdot[1S + 1M] \tag{5.5}$$

לכן, זמן החישוב הכולל הוא:

$$.t^h_{op} \doteq N^2 \cdot \left\{\frac{I}{B^2} \cdot [4S + 1M] + \frac{4}{3} \cdot [1S + 1M]\right\} \tag{5.6}$$

בדרך כלל B^2 גדול בהרבה ממספר האיטרציות I (ערכים אופייניים הם $B^2 =$ 64, $I < 8$), כך שהאיבר הראשון ניתן להזנחה, ומתקבל:

$$.t_{op}^{h} \doteq N^2 \cdot \frac{4}{3} \cdot [1S + 1M] \qquad (5.7)$$

לדוגמא, אם נניח $M = k \cdot S$, נמצא שהיחס בין מספרי הפעולות הוא:

$$.Q \triangleq \frac{t_{op}^{h}}{t_{op}^{i}} = \frac{N^2 \cdot \frac{4}{3} \cdot S \cdot [1+k]}{N^2 \cdot I \cdot S \cdot [4+k]} = \frac{4 \cdot [1+k]}{3 \cdot I\,[4+k]} \qquad (5.8)$$

(א) עבור מעבד נקודה-צפה (floating point processor), ניתן להניח $k = 1$, ולאחר הצבה נקבל

$$,Q = \frac{8}{15 \cdot I} \qquad (5.9)$$

אשר מביא לחיסכון בסדר גודל אם $I \geq 6$.

(ב) עבור מעבד נקודה-קבועה (fixed point processor), נניח לדוגמא $k = 8$, ולאחר הצבה נקבל

$$.Q = \frac{1}{I} \qquad (5.10)$$

ניתן לראות שככל שמספר האיטרציות הנחוץ להגיע לנקודת שבת, I, יותר גדול, גדל יתרונה של שיטת הפיענוח המוצעת. מספר האיטרציות נקבע בד"כ לפני הקידוד, והוא תלוי בעיקר בכיווציות הצפויה של הטרנספורמציה וברמת הדיוק הדרושה.

נעיר שעל-פי מה שציינו בסעיף 4.2.2 (ראה גם [48]), קיימת שיטת קידוד המשלבת את שיטת הקידוד שהוצגה עד-כה עם הורדת רכיב ה-.D.C של בלוקי הטווח (כלומר, הפחתת הערך הממוצע של הבלוק). לשיטה זו התכונות הבאות הנוגעות לענייננו:

1. ניתן למצוא את הרמה העליונה בפירמידת נקודות-השבת ללא צורך באיטרציות, אלא ישירות מתוך רכיבי ההסחה של הקוד. תכונה זו שימושית לצורך פיענוח היררכי.
2. לצורך פיענוח איטרטיבי, ניתן להסתפק במספר קטן של איטרטציות כדי להגיע בדיוק לנקודת השבת [49]. מספר האיטרציות המדויק תלוי בפרמטרי הקידוד, אולם מספר אופיני הוא 3 , בניגוד ל-8 בשיטה עם איטרציות.

ניתן לראות ששיטה זו אינה מביאה לשינוי מהותי בביטויים שהוצגו (למשל, משוואות (5.9) ו- (5.10) נשארות אותן), אולם ערך אופיני למספר האיטרציות, I, יהיה כעת $I = 3$.

5.2 אינטרפולציה פרקטלית של האות

אינטרפולציה של אות עוסקת במציאת ערכים של האות בנקודות ביניים נוספות על אלו הנתונות. אנו נעסוק בעיקר בהשגת רזולוציה גבוהה יותר [59],כלומר רזולוצית-על, כאשר נתונים ערכי האות ברזולוציה מסוימת. למשל, תהי $g(x) \in L^\infty[0,1]$ פונקציה שאינה ידועה לנו, אולם נתון לנו הוקטור $\vec{v}^1$, שאיבריו הם $\vec{v}^1(i) = g_N(i), i = 1, 2, \cdots, N$ (ראה הגדרת $g_N(i)$ בפרק 3.2). דהיינו, הוקטור מתאר את האות $g(x)$ ברזולוציה N. מטרתנו היא מציאת וקטור $\vec{v}^2$, בעל $2N$ איברים, אשר יהווה קירוב לאות $g(x)$ ברזולוצייה $2N$, דהינו $g_{2N}(i), i = 1, 2, \cdots, 2N$. תהליך המעבר מרזולוציה נתונה לרזולוציה גבוהה יותר נקרא גם 'זום' (zoom-in או בפשטות zooming).

מתוך המבנה הפירמידלי שהוצג, נובעת שיטה ישירה לביצוע 'זום' על אות נתון. בהינתן

נקודת-שבת $\vec{f}^1$, שהיא וקטור באורך N (שהוא האות $g(x) \in L^\infty[0,1]$ ברזולוציה N, כאשר $g(x)$ לא ידועה), מציאת קוד ה-IFS של הוקטור מאפשרת לבנות את המבנה ההיררכי, כלומר את פירמידת נקודות-השבת. קוד ה-IFS מתאר גם קשר ישיר בין שתי רמות סמוכות בפירמידה (משפט הקשר-ההיררכי, משוואות (3.1) - (3.2)). לפיכך, לאחר מציאת קוד ה-IFS של $\vec{f}^1$, כל שיש לעשות כדי לקבל וקטור באורך $2N$ הוא להפעיל את אלגוריתם ה-zoom-in על $\vec{f}^1$, כפי שתואר במשוואה (3.2), ומתקבל הוקטור החדש $\vec{f}^2$. הוקטור $\vec{f}^2$ מהווה חלק אינטגרלי מפירמידת נקודות-השבת, והוא נקודת-שבת של קוד ה-IFS כאשר משתמשים ב-$B = 2B_1$. וקטור זה יכול לשמש כקירוב להצגה ברזולוציה יותר גבוהה, כלומר קירוב לאות g_{2N}.

5.2.1 מימד פרקטלי של האות

הפונקציה המוכלת בקוד כבר הוצגה מקודם, ולפי האמור לעיל, ניתן לראות בה מקרה גבול של אינטרפולציה לרזולוציה גבוהה יותר ויותר. לכן, תכונות המתארות את 'אופיה' של פונקציה זו יכולות לסייע להבין את אופי האינטרפולציה המתקבלת. תכונה אחת כזו היא המימד הפרקטלי של הפונקציה המוכלת בקוד.

משפט 4 (**מימד פרקטלי של הפונקציה המוכלת בקוד**) יהי נתון קוד-IFS שנסמנו W, ויהיו $[F], [A], [D]$, ו-$\vec{b}$ הייצוג המטריצי שלו, כאשר משתמשים ב-$B = 1$ (ראה פרק 4), כלומר:

$$W(\vec{v}) = [F]\vec{v} + \vec{b} = ([A]\frac{1}{2}[D])\vec{v} + \vec{b}. \qquad (5.11)$$

תהי $G(x)$ הפונקציה המוכלת בקוד המתאימה ל-W. המימד הפרקטלי של $G(x)$ מקיים

$$1 \leq \mathcal{D} \leq max(1, 1 + log_2(\lambda)), \qquad (5.12)$$

כאשר λ היא הערך העצמי הממשי הגדול ביותר של המטריצה $(||[A]|| \cdot [D])$, ו-$||[A]||$ היא המטריצה אשר איבריה הם הערכים המוחלטים של האיברים המתאימים במטריצה $[A]$.

הוכחת המשפט נמצאת בנספח D.

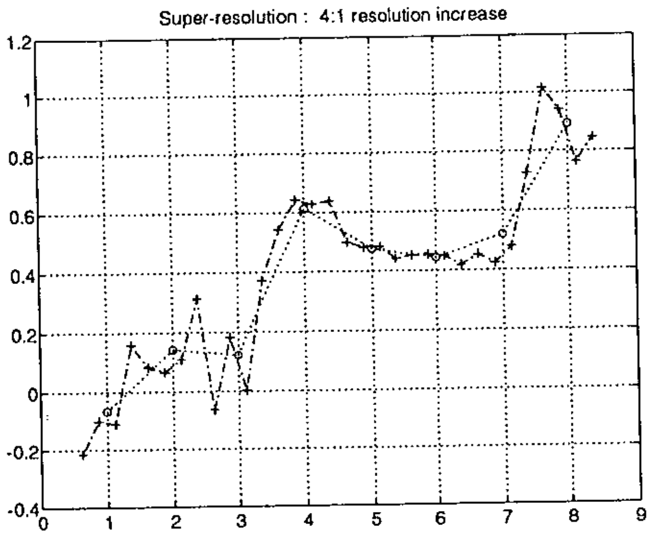
חשיבותו של המימד הפרקטלי נובעת מכך שהוא מתאר לנו את 'אופי' הוקטורים המתקבלים בתהליך האינטרפולציה. (ראה [4] עבור דיון מפורט על פונקציות אינטרפולציה פרקטליות). ע"י אילוץ תנאים שונים, בזמן תהליך הקידוד, על $\frac{D}{B}$ ועל $|a_i|$, ניתן לשנות את המימד הפרקטלי המתקבל של הפונקציה המוכלת בקוד (ראה גם בחלק הדן בכיוונים להמשך המחקר).

לסיכום, הראינו עד כה ששיטת האינטרפולציה המוצעת יוצרת פונקציה (וקטורים) עם מימד פרקטלי, אשר חסם עליו ניתן לחישוב ישירות מתוך קוד ה-IFS.

דוגמא ליישום השיטה לרזולוצית-על ולאופי המיוחד שלה מובאת באיור 5.1. וקטור באורך $N_1 = 256$ משמש כוקטור המקורי, דהיינו $\vec{v}^1$. וקטור זה משמש למציאת קוד IFS, בעל נקודת-שבת $\vec{f}^1$. החסם על המימד הפרקטלי של הפונקציה המוכלת בקוד חושב (בעזרת משפט 4) והוא $\mathcal{D} = 1.16$. בעזרת קוד ה-IFS חושב הוקטור $\vec{f}^4$ באורך $4 \cdot N_1 = 1024$. באיור מובאים 32 האיברים הראשונים של $\vec{f}^4$ ('+' מחוברים עם קו מקווקד), וכמו כן מובאות 8 הנקודות המקוריות של $\vec{v}^1$ עם אינטרפולציה לינארית ביניהן ('o' מחוברים עם נקודות).

ניתן להבחין במאפיינים הבאים של האינטרפולציה:

1. הממוצע של 4-איברים סמוכים של $\vec{f}^4$ הוא בקירוב האיבר המתאים של $\vec{v}^1$ (שוויון מדויק יתקיים אם קוד ה-IFS הוא קוד ללא-עיוות, כלומר $\vec{f}^1 = \vec{v}^1$).



איור 5.1: רזולוצית-על פרקטלית ('+' וקו מקווקד) ואינטרפולציה-לינארית ('o' ונקודות).

Fig. 5.1: Super-resolution via IFS-code ('+' and a dash-dot line) .vs. linear-interpolation ('o' and a dotted line).

2. פונקצית האינטרפולציה הפרקטלית אינה עוברת בהכרח דרך הנקודות הנתונות ע"י $\vec{v}^1$. יתרה מזו, אם נחשב את $\vec{f}^8$, היא לא בהכרח תעבור דרך הנקודות של $\vec{f}^4$.

3. אינטרפולציה לינארית נוטה להחליק את הפונקציה, בעוד שאינטרפולציה פרקטלית שומרת על מימד פרקטלי מסוים, דבר אשר מבטיח עושר פרטים גם ברזולוציות גבוהות. תכונה זו באה לידי ביטוי בעיקר כאשר דנים במרקמים טבעיים (textures): בעוד שאינטרפולציה לינארית יוצרת בד"כ תמונה מטושטשת, הרי שאינטרפולציה פרקטלית שומרת על מראה 'מחוספס'. דוגמאות לביצוע אינטרפולציה על תמונות מודגם בנספח (E) שם מודגם האלגוריתם על תמונה המכילה הן מרקם (הרקע) והן גופים עם שפות חדות (היד,מחבט).

5.2.2 אינטרפולציה בגורמים רציונליים

אחרי תיאור השיטה לקבלת רזולוצית-על פרקטלית, מתעוררת השאלה הבאה:

> לפי השיטה שתוארה, ניתן לחשב אך-ורק רזולציות של האות שהן מכפלה של N (אורך וקטור מקורי) בחזקות של 2 כלומר $N, 2N, 4N, 8N, \cdots$. האם ניתן לחשב גם רזולציות אחרות, למשל $\frac{3}{2}N$?

התשובה לשאלה זו מוצגת בסעיף זה.

בתאור תהליך הפיענוח של קוד IFS (2.2.2), מתוארת האפשרות להחליף את הערך המקורי של $B = B_1$ בערך אחר. בבנית הפירמידה של נקודות-שבת, כפי שתוארה לצורך קבלת רזולוצית-על, הערך החדש של B נלקח להיות $B_2 = 2B_1$. ערך זה מוביל לנקודת שבת באורך $N_2 = 2N_1 = 2B_1M_R$. אולם, נניח כי בוחרים את הערך $B = B_{\frac{B_1+1}{B_1}} = B_1 + 1$. ערך זה מוביל לנקודת-שבת באורך

$$N_{\frac{B_1+1}{B_1}} = B_{\frac{B_1+1}{B_1}} \cdot M_R = N_1 + M_R. \tag{5.13}$$

באופן דומה לקיחת $B_{\frac{B_1+2}{B_1}} = B_1 + 2$ מובילה לאורך של

$$N_{\frac{B_1+2}{B_1}} = B_{\frac{B_1+2}{B_1}} \cdot M_R = N_1 + 2M_R. \tag{5.14}$$

אם נמשיך באותה דרך, אנו רואים כי ניתן לקבל מגוון רחב של רזולציות, מכומתות בצעדים של M_R. נקודות השבת המתקבלות יוצרות מבנה פירמידלי, אשר יקרא פירמידה רציונלית (*rational-pyramid*). כל רמה בפירמידה הרציונלית היא נקודת-שבת של הקוד, והפירמידה של נקודות-שבת מוכלת בפירמידה הרציונלית.

לדוגמא, איור 5.2(א)-(ג) מדגים פיענוח של קוד ה-IFS הנתון באיור 2.1, עם $B = 4$, $B = 3$, ו-$B = 2$, בהתאמה. במקרה זה, $M_R = 4$.

22	18	10	14	6	2	14	10

(ג)

22	19	18	11	12	13	6	5	2	14	11	10

(ב)

23	21	17	19	11	9	15	13	5	7	3	1	15	13	9	11

(א)

איור 5.2: פיענוח עם (א) $B = 4$, (ב) $B = 3$ (מקורב לערך השלם הקרוב), ו-(ג) $B = 2$.

Fig. 5.2: Decoding with (a) $B = 4$, (b) $B = 3$ (approximated to the nearest integer value), and (c) $B = 2$.

ניתן להראות מידית, בעזרת הייצוג המטריצי ומשפט הפונקציה המוכלת בקוד, שכאשר נתון קוד IFS ורמה אחת בפירמידה הרציונלית, אזי ניתן לחשב כל רמה אחרת בפירמידה, למרות שלא בהכרח באופן ישיר (מובן שניתן לחשב כל רמה בפירמידה בעזרת הקוד בלבד, אולם כאן נראה כיצד ניתן להגיע לכל רמה ע"י מעבר בין רמות).

לדוגמא, בהינתן קוד IFS וקטור $\vec{f}^1$ המתאים לערך של $B = B_1 = 3$, מהו האלגוריתם למציאת $\vec{f}^{\frac{4}{3}}$, המתאים לערך של $B = 4$? הדרך היא לחשב את $\vec{f}^2$ מתוך $\vec{f}^1$ והקוד, לפי (3.2). $\vec{f}^2$ מתאים בדוגמא זו לערך של $B = 6$. בשלב הבא, נחשב את $\vec{f}^4$ באותה דרך, כאשר $\vec{f}^4$ מתאים לערך של $B = 12$. כעת, כל איבר של $\vec{f}^{\frac{4}{3}}$ ניתן לחישוב ישיר ע"י מיצוע כל 3 איברים סמוכים של $\vec{f}^3$ (נובע ישירות מתוך משפט הפונקציה המוכלת בקוד). סיכום השיטה מוצג בנוסחאות הבאות (ראה גם (3.10)):

$$\vec{f}^1(k) = \frac{1}{4}(\vec{f}^4(4k) + \vec{f}^4(4k-1) + \vec{f}^4(4k-2) + \vec{f}^4(4k-3)) \qquad (5.15)$$

$$\vec{f}^{\frac{4}{3}}(k) = \frac{1}{3}(\vec{f}^4(3k) + \vec{f}^4(3k-1) + \vec{f}^4(3k-2)) \qquad (5.16)$$

5.3 שיטות דגימה שונות

אותות בעולם הטבע הם בדרך כלל פונקציות של משתנה רציף. כך, למשל, אות דיבור הוא פונקציה של משתנה רציף (זמן), ותמונה היא פונקציה של שני משתנים רציפים ((x,y)). אולם, כאשר אנו מעבדים אותות בעזרת מחשב, אנו מבצעים דגימה של האות, ובכך עוברים לעסוק בוקטורים (או מטריצות). כפי שראינו, גם בייצוג ההירכי קיימת פונקציה של משתנה רציף (הפונקציה המוכלת בקוד) וקיימים וקטורים (נקודות השבת ברזולוציות השונות). בסעיף זה נתאר את הקשר הקיים בין צורות דגימה שונות, הפונקציה המוכלת בקוד ואופן חישוב הקוד.

ראשית, נזכיר את צורת הדגימה בה נתקלנו כבר. בהינתן פונקציה $G(x)$, שהיא הפונקציה המוכלת בקוד של קוד IFS כלשהוא, נסמן ב-$g_N(i), i = (1, \cdots, N)$, את דגימתה ע"י אינטגרציה:

$$.g_N(i) \triangleq N \int_{(i-1)\frac{1}{N}}^{i\frac{1}{N}} G(x)\, dx, \qquad i = (1, \cdots, N) \tag{5.17}$$

לפי משפט הפונקציה המוכלת בקוד, $g_N(i)$ שהוגדרה לעיל היא נקודת שבת של קוד ה-IFS, תוך שימוש בפונקצית כיווץ-מרחבית φ המבצעת מיצוע ודצימציה, או כפי שנכתב במשוואה (2.16):

$$.\varphi(\vec{d_l})(j) = \frac{1}{2}(\vec{d_l}(2j) + \vec{d_l}(2j-1)) \tag{5.18}$$

קידוד של g_N המשתמש בפונקציה φ זו, יכול למצוא את קוד ה-IFS הנכון (כלומר, את הקוד עם פונקציה מוכלת בקוד הזהה ל-$G(x)$), ובמקרה זה הקוד יהיה ללא עיוות. קיימים מקרים שבהם יש יותר מקוד IFS אחד בעל אותה נקודת שבת ברזולוציה N, ובמקרים אלה ייתכן שלקודים שונים מתאימה פונקציה מוכלת בקוד שונה. מקרים אלה אינם תורמים לדיון כאן, ולכן נתעלם מהם בהמשך.

אולם, נניח כעת שהפונקציה $G(x)$ נדגמה באופן שונה, שנקרא לו דגימה-נקודתית (*point-sampling*), ולכן $g^s_N(i)$ נתון באופן הבא:

$$.g^s_N(i) \triangleq G((i-1)\frac{1}{N}), \qquad i = (1, \cdots, N) \tag{5.19}$$

ניתן להראות (ראה הסתיגות מיד) שכל התוצאות הקודמות מתקיימות אם נשנה את פונקצית הכיווץ-המרחבי באופן הבא:

$$.\varphi^s_N(\vec{d}_l)(j) \triangleq \vec{d}_l(2j-1) \tag{5.20}$$

לפני שנפרט, רצוי להעיר שכל הטיפול במקרה הדגימה הנקודתית מוצג כאן ללא דיוק מתמטי מלא. לצורך הדגמה, ננסח שוב מספר משפטים והגדרות עבור המקרה של דגימה-נקודתית (ההוכחות של המשפטים דומות מאד להוכחות הקודמות, ולכן הושמטו):

- **משפט 5 (קשר הירכי)** יהי נתון קוד IFS, אשר מוביל ל-W^1 עם $B = B_1$, ומוביל ל-$W^{\frac{1}{2}}$ עם $B = B_1/2$. נקודות השבת של הטרנספורמציות האלה יסומנו ב-$\vec{f}^1$ ו-$\vec{f}^{\frac{1}{2}}$, בהתאמה. אזי:

ירידה ברזולוציה (Zoom-out) :

$$\vec{f}^{\frac{1}{2}}(j) = \vec{f}^{1}(2j-1), \qquad j = (1, \cdots, M_R(B_1/2)) \tag{5.21}$$

עלייה ברזולוציה (Zoom-in) :

$$\vec{f}^{1}((i-1)B_1 + j) = a_i \vec{f}^{\frac{1}{2}}((m_i - 1)D_h{}^{\frac{1}{2}} + j) + b_i \tag{5.22}$$

$$i = (1, \cdots, M_R), \qquad j = (1, \cdots, B_1)$$

כאשר $D_h{}^{\frac{1}{2}} \triangleq \frac{D_h{}^1}{2} = \frac{B_1}{2}$.

- **הגדרה 2** תהי נתונה פונקציה $G(x) \in L^\infty\,[0,1]$, נגדיר את $G^s_r(i)$ באופן הבא:

$$G^s_r(i) \triangleq G((i-1)\frac{1}{r}), \qquad i = (1, \cdots, r) \tag{5.23}$$

$G^s_r(i)$ נקראת הפונקציה $G(x)$ ברזולוציה דגימה-נקודתית r. אנו אומרים ש-$G^s_{r_1}(i)$ היא עדינה יותר (כלומר, עם רזולוציה גבוהה יותר,(finer)) מאשר $G^s_{r_2}(i)$ שהיא גסה יותר (coarser) אם $r_1 > r_2$.

- **משפט** 6 (**הפונקציה המוכלת בקוד**) בהינתן קוד IFS, קיימת פונקציה אחת ויחידה $G(x) \in L^\infty\,[0,1]$, כך שהוקטור $\vec{f}^s_N \in \Re^N$ הוא נקודת שבת של קוד ה-IFS (תוך שימוש ב-φ^s) אם ורק-אם הוא שווה לפונקציה $G(x)$ ברזולוציה דגימה-נקודתית $N = r$ לכל N, כלומר,

$$\vec{f}_N(j) = G^s_N(j), \qquad j = 1, 2, \cdots, N \tag{5.24}$$

הפונקציה $G(x)$ נקראת <u>הפונקציה המוכלת בקוד</u>[1] (<u>IFS embedded function</u>).

החשיבות בניסות וטיפול בשיטות דגימה שונות נובע מכך שבדרך-כלל בעית הקידוד ניתנת להיכתב באופן הבא:

> נתון מודל עבור האות בטבע, למשל אות דיבור או תמונה, שהוא פונקציה של משתנה רציף. המודל קובע כי האות הוא בעל אופי פרקטלי. אות זה עובר תהליך של דגימה (כדי לעבדו באופן סיפרתי), ומתוך האות הדיסקרטי יש לחשב את הקוד.

---

[1] הסיבה לשם הזהה עבור שני המקרים של דגימות שונות יוסבר בהמשך

כפי שהראינו, הדרך הנכונה למציאת הקוד, כלומר הדרך שתוביל למציאת הקוד המתאר את האות הרציף, תלויה בשיטת הדגימה. לכן, בהינתן מודל מתאים ושיטת דגימה, ניתן להשתמש בשיטת הקידוד המתאימה, דהיינו פונקציית כיווץ מרחבי מתאימה.

חשוב לציין שהפונקציה המוכלת בקוד, בשתי שיטות הדגימה, היא אותה. עובדה זו אינה מפתיעה, משום שככל ש-N גדל, $g_N^s(i)$ ו-$g_N(i)$ מתקרבים, כאשר

$$g_N^s(i) \triangleq G((i-1)\frac{1}{N}) \tag{5.25}$$

וגם

$$.g_N(i) \triangleq N \int_{(i-1)\frac{1}{N}}^{i\frac{1}{N}} G(x)\, dx \tag{5.26}$$

פרק 6

חסם משופר לצורך קידוד

כפי שראינו עד כה, משפט הפסיפס (Collage-Theorem) מתאר חסם על שגיאת המקודד, וחסם זה משמש כמוטיבציה לאלגוריתם הקידוד. בפרק זה נתאר חסם הדוק יותר מזה המוצג במשפט הפסיפס. חסם זה עושה שימוש בהצגה של האות במספר רזולוציות שונות. יש לציין שחסם זה מותאם אך-ורק לטרנספורמציות המשתמשות בפונקציות כיווץ-מרחבי הפועלות בעזרת מיצוע איברים סמוכים ודצימציה.

6.1 משפט-פסיפס משופר

לצורך הצגת המשפט, נגדיר מספר הגדרות חדשות (ונחזור על מספר הגדרות קודמות, לשם בהירות):

1. יהי נתון קוד IFS המוביל ל-W^1 עם $B = B_1$, ומוביל ל-$W^{1/2}$ עם $B = B_1/2$. באופן דומה נסמן ב-$W^{1/2^k}$ את הטרנספורמציה המשתמשת ב-$B = B_1/2^k$. אם לא יסומן גודל על W, אזי הכוונה היא ל-W^1. כמו כן, הפעלה של W k-פעמים, תסומן $W^{\circ k}$.

2. יהי $\vec{x} \in \Re^N$ וקטור באורך N. נסמן כ-$(\vec{x})_{2^l}$ את האות המתקבל ע"י שכפול כל איבר של $\vec{x}$ 2^l פעמים, דהיינו:

$$(\vec{x})_{2^l} \in \Re^M, \qquad M = 2^l \cdot N, \qquad l = 0,1,2,\ldots.$$

לשם דוגמא,

$$\begin{array}{lll} \vec{x} = 1,3,2 & N = 3 & \\ (\vec{x})_1 = 1,3,2 & M = 3 & l = 0 \\ (\vec{x})_2 = 1,1,3,3,2,2 & M = 6 & l = 1 \end{array}$$

3. (חזרה על סימון קודם) נסמן ב-$\vec{x}^{1/2}$ את האות המתקבל לאחר מיצוע איברים סמוכים ב-$\vec{x}$ ודצימציה. באופן דומה יסמן $\vec{x}^{1/2^k}$ את האות לאחר מיצוע 2^k איברים סמוכים, ודצימציה מתאימה (של 2^k).

4. נסמן כ-$d_{1/2^k}(\cdot,\cdot)$ פונקצית מרחק הפועלת במרחב $\Re^{N/2^k}$. לצורך מעבר בין רמות רזולוציה שונות, דבר שנזדקק לו מיד, נדרוש שפונקצית המרחק במימדים השונים תקיים את התכונה הבאה:

$$d_{1/2^k}(\vec{x}^{1/2^k},\vec{y}^{1/2^k}) = d_{1/2^l}((\vec{x}^{1/2^k})_{2^{(k-l)}},(\vec{y}^{1/2^k})_{2^{(k-l)}}), \qquad \forall\, 0 \le l \le k \qquad (6.1).$$

פונקציות מרחק המקיימות תנאי זה הן, למשל, d_∞, ויתר המרחקים המסתמכים על נורמות l^p לאחר נירמול. למשל, הגדרת l^2 מנורמל היא:

$$d^{l^2}_{1/2^k}(\vec{x},\vec{y}) \triangleq (\frac{1}{N/2^k}\sum_{i=1}^{i=N/2^k}(x_i-y_i)^2)^{1/2}.$$

משפט 7 (**משפט-פסיפס משופר**) יהי נתון קוד IFS שנסמנו $W : \Re^N \to \Re^N$. הקוד פועל עם בלוקי-טווח בגודל B. נסמן את מקדם הכיווץ של $W^{\circ k}$ כ-s_k (כאשר $s_0 = 1$, $W^{\circ 0} = I$). נסמן

ב-$\vec{f}$ את וקטור נקודת-השבת של W. אזי,

$$.d(\vec{x},\vec{f}) \leq \sum_{k=0}^{log_2(B)-1} s_k \cdot d_{1/2^k}(\vec{x}^{1/2^k}, W^{1/2^k}\vec{x}^{1/2^k}) \qquad (6.2)$$

לפני ההוכחה, נסביר מספר פרטים, אשר יעזרו בהבנת המשפט וההוכחה:

1. נדגים מקרה פרטי של הוקטורים השונים.

$\vec{x}$	=	0	2	4	6
$\vec{x}^{1/2}$	=	1		5	
$(\vec{x}^{1/2})_2$	=	1	1	5	5
$\vec{x}^{1/4}$	=	3			
$(\vec{x}^{1/4})_2$	=	3		3	
$(\vec{x}^{1/4})_4$	=	3	3	3	3

2. כזכור, הטרנספורמציה W הפועלת על וקטור, מפעילה תחילה כיווץ מרחבי על בלוקי-תחום. פעולה זו של כיווץ מרחבי מורכבת ממיצוע שני איברים סמוכים, ולאחר מכן דצימציה (דבר שמקטין את אורך בלוק-תחום D לאורך של בלוק-טווח B). לכן, פונקצית הכיווץ-המרחבי, וגם הטרנספורמציה W, רגישות רק לממוצע של זוגות איברים סמוכים.

מתוך כך מתקבלות שלוש טענות העזר הבאות (יש לשים לב שטענות אלה נכונות עבור כל וקטור $\vec{x} \in \Re^N$) :

(א) טענה זו מהווה בסיס ליתר הטענות,

$$.W^{\circ k}\vec{x} = W^{\circ k}(\vec{x}^{1/2^k})_{2^k}, \qquad k = 1, \ldots, log_2(B) \qquad (6.3)$$

הסבר הטענה:

עבור $k = 1$ מתקבלת המשוואה $W\vec{x} = W(\vec{x}^{1/2})_2$. עובדה זו ברורה מידית מתוך העובדה ש-W מתחשבת רק בממוצע של איברים סמוכים בוקטור עליו היא מופעלת, ולכן $(\vec{x}^{1/2})_2$ וגם $\vec{x}$ יובילו לאותה תוצאה.

עבור $k = 2$ מתקבלת המשוואה $W^{\circ 2}\vec{x} = W^{\circ 2}(\vec{x}^{\frac{1}{4}})_4$. לצורך הסבר זה, נשים לב לעובדה ש-$W^{\circ 2}$ מושפע רק מממוצע של 4 ערכים עוקבים בוקטור הכניסה. כלומר, כל איבר בתוצאה מושפע מ-4 איברים סמוכים בוקטור הכניסה, אשר עברו תהליך כפול של כיווץ מרחבי.

לפי אותה דרך, ניתן להמשיך את ההוכחה עבור ערכי k אחרים.

יש לשים לב שהערך המקסימלי של k שעבורו ההוכחה תקפה הוא $k = log_2(B)$. מעבר לערך זה, המיצוע שיתקבל ב-$\vec{x}^{1/2^k}$ יתפרש על יותר מידי איברים (מעבר לתחום של בלוק-תחום מסוים), ולכן ההוכחה תיכשל.

(ב) זהו ניסוח כללי יותר לטענה המובאת ב-א, והוכחתו מידית מתוך הוכחת א.

$$.W^{\circ k}\vec{x} = W^{\circ k}(\vec{x}^{1/2^l})_{2^l}, \qquad k = 1, \ldots, log_2(B), \qquad 0 \leq l \leq k \qquad (6.4)$$

(ג) טענה נוספת, שאף היא נעזרת בטענה א,

$$.W^{\circ k}\vec{x} = W^{\circ (k-1)}(W^{1/2^{(k-1)}}\vec{x}^{1/2^{(k-1)}})_{2^{(k-1)}}, \qquad k = 2, \ldots, log_2(B) \qquad (6.5)$$

ראשית, נוכיח טענה זו עבור $k = 2$,

$$W^{\circ 2}\vec{x} = W(W^{1/2}\vec{x}^{1/2})_2. \qquad (6.6)$$

כדי להוכיח ביטוי זה, מספיק להראות שהערך של ממוצע איברים סמוכים ב-$W\vec{x}$, שווה לערך הממוצע ב-$(W^{1/2}\vec{x}^{1/2})_2$. כדי להראות זאת, נסתכל על הפעולות המבוצעות על רביעיית איברים שכנים ב-$\vec{x}$, ועל האיברים המתאימים ב-$\vec{x}^{1/2}$.

$$\begin{aligned} \vec{x} &= [a, b, c, d, \dots \\ \vec{x}^{1/2} &= [\tfrac{a+b}{2}, \tfrac{c+d}{2}, \dots \end{aligned}$$

לאחר כיווץ מרחבי, 'עוברים' איברים אלה, המהווים חלק מבלוק-תחום, למיקום אחר בוקטור, באופן הבא (נניח גורם-כיווץ והסחה a_i ו-b_i בהתאמה):

$$\begin{aligned} W\vec{x} &= [\dots, \tfrac{a+b}{2} \cdot a_i + b_i, \tfrac{c+d}{2} \cdot a_i + b_i, \dots \\ W^{1/2}\vec{x}^{1/2} &= [\dots, \tfrac{a+b+c+d}{2} \cdot a_i + b_i, \dots \end{aligned}$$

כעת, הוקטור $W^{1/2}\vec{x}^{1/2}$ עובר שיכפול איברים, ולכן הממוצע של שני איברים סמוכים, בשני המקרים, הוא אותו.

עבור $k = 3$, ההוכחה דומה, אלא שהפעם יש להסתפק בהוכחה שהממוצע של 4 איברים עוקבים הוא אותו דבר עבור $(W^{1/4}\vec{x}^{1/4})_4$ ועבור $(W\vec{x})$.

3. $\vec{f}$ הוא נקודת-שבת, ולכן

$$W\vec{f} = \vec{f}.$$

4. כפי שציינו בפרק 4.2.2 (ראה גם [51]), עבור הרמה העליונה (Top-Level) בפירמידה מתקיים,

$$d_{1/B}(\vec{x}^{1/B}, \vec{f}^{1/B}) = 0. \qquad (6.7)$$

דהיינו, ברמה העליונה ערך כל איבר הוא ממוצע ערכי הבלוק-טווח המתאים, גם בפירמידה הנובעת מנקודות-השבת, וגם בפירמידה הנובעת מהאות עצמו.

כעת, נחזור להוכחת המשפט.

הוכחה : תוך שימוש באי-שוויון המשולש, ובעובדה ש-$\vec{f}$ היא נקודת-שבת של W, נקבל,

$$\begin{aligned} d(\vec{x},\vec{f}) &= d(\vec{x}^1,\vec{f}^1) \\ &\leq d(\vec{x}^1, W^{\circ 1}\vec{x}^1) + d(W^{\circ 1}\vec{x}^1, W^{\circ 2}\vec{x}^1) + d(W^{\circ 2}\vec{x}^1, W^{\circ 3}\vec{x}^1) + \\ &\quad + \ldots + d(W^{\circ\, log_2(B)}\vec{x}^1, W^{\circ\, log_2(B)}\vec{f}^1) \end{aligned}$$

כעת, נסתכל על האיברים השונים בסכום זה, ונתחיל מהאיבר השני,

$$.d(W^{\circ 1}\vec{x}^1, W^{\circ 2}\vec{x}^1) \tag{6.8}$$

לפי (6.3),

$$.W^{\circ 1}\vec{x}^1 = W^{\circ 1}(\vec{x}^{1/2})_2 \tag{6.9}$$

לפי (6.5),

$$,W^{\circ 2}\vec{x}^1 = W^{\circ 1}(W^{1/2}\vec{x}^{1/2})_2 \tag{6.10}$$

ולכן,

$$\begin{aligned} d(W^{\circ 1}\vec{x}^1, W^{\circ 2}\vec{x}^1) &= d(W^{\circ 1}(\vec{x}^{1/2})_2, W^{\circ 1}(W^{1/2}\vec{x}^{1/2})_2) && (6.11) \\ &\leq s_1 d((\vec{x}^{1/2})_2, (W^{1/2}\vec{x}^{1/2})_2) && (6.12) \\ &= s_1 d_{1/2}(\vec{x}^{1/2}, W^{1/2}\vec{x}^{1/2}) && (6.13) \end{aligned}$$

באופן דומה לגבי האיבר השלישי, נקבל,

$$d(W^{\circ 2}\vec{x}^1, W^{\circ 3}\vec{x}^1) = d(W^{\circ 2}(\vec{x}^{1/4})_4, W^{\circ 2}(W^{1/4}\vec{x}^{1/4})_4) \quad (6.14)$$

$$\leq s_2 d((\vec{x}^{1/4})_4, (W^{1/4}\vec{x}^{1/4})_4) \quad (6.15)$$

$$= s_2 d_{1/4}(\vec{x}^{1/4}, W^{1/4}\vec{x}^{1/4}) \quad (6.16)$$

המשך באופן דומה, ושימוש בעובדה שברמה העליונה המרחק בין הביטויים השונים הוא אפס, מוביל לתוצאה.

(לגבי חישוב מקדמי הכיווץ s_k, ראה פרק 4 וגם [51]).

מ.ש.ל. ■

<u>הערה :</u>

ניתן גם להוכיח את החסם הבא, שהוא פחות הדוק מהקודם,

$$.d(\vec{x}, \vec{f}) \leq \sum_{k=0}^{log_2(B)-1} s_k \cdot d((\vec{x}^{1/2^k})_{2^k}, W(\vec{x}^{1/2^k})_{2^k}) \quad (6.17)$$

לצורך הוכחת חסם זה נדגים רק את קבלת האיבר עבור $k = 1$, מתוך האיבר המתאים בסכום שתואר בהוכחה מקודם (6.8) ,

$$.d(W^{\circ 1}\vec{x}^1, W^{\circ 2}\vec{x}^1) \quad (6.18)$$

לפי (6.3) ,

$$,W^{\circ 1}\vec{x}^1 = W^{\circ 1}(\vec{x}^{1/2})_2 \quad (6.19)$$

ושוב, לפי (6.4) ,

$$,W^{\circ 2}\vec{x}^1 = W^{\circ 2}(\vec{x}^{1/2})_2 \quad (6.20)$$

ולכן,

$$d(W^{\circ 1}\vec{x}^1, W^{\circ 2}\vec{x}^1) = d(W^{\circ 1}(\vec{x}^{1/2})_2, W^{\circ 2}(\vec{x}^{1/2})_2) \le \quad (6.21)$$

$$s_1 d((\vec{x}^{1/2})_2, W^{\circ 1}(\vec{x}^{1/2})_2) \quad (6.22)$$

ואת יתר האיברים בסכום מקבלים באופן דומה.

<u>סוף הערה.</u>

6.2 דיון בחסם המשופר

ראשית, נכתוב שוב, בצורה תואמת, את שני החסמים:

- חסם פסיפס קלאסי,

$$d(\vec{x}, \vec{f}) \le \sum_{k=0}^{\infty}(s^k \cdot d(\vec{x}, W\vec{x})) \quad (6.23)$$

כאשר $s \triangleq s_1$ הוא מקדם הכווץ של W, ו-s^k פרושו "s בחזקת k".

- חסם פסיפס משופר,

$$d(\vec{x}, \vec{f}) \le \sum_{k=0}^{log_2(B)-1} s_k \cdot d_{1/2^k}(\vec{x}^{1/2^k}, W^{1/2^k}\vec{x}^{1/2^k}) \quad (6.24)$$

כאשר s_k הוא מקדם הכווץ של $W^{\circ k}$.

בצורת כתיבה זו, ניתן להבחין בגורמים לכך שהחסם החדש הדוק יותר:

<u>מקדמי כיווץ</u> - $s_k \le s^k$ (ראה מיד). כלומר, עבור החסם המשופר המקדמים של איברים דומים בסכום הינם קטנים יותר או שווים למקדמים הרגילים.

הסבר: לפי הגדרת מקדם הכיווץ,

$$\begin{aligned} d(W^{\circ k}x, W^{\circ k}y) &\le s \cdot d(W^{\circ k-1}x, W^{\circ k-1}y) \\ &\le \ldots \le s^k \cdot d(x,y) \end{aligned}$$

מאידך, s_k הוא המספר המינימלי המקיים

$$d(W^{\circ k}x, W^{\circ k}y) \le s_k \cdot d(x,y)$$

ולכן $s_k \le s^k$.

מרחקים שונים - נשווה בין ביטויי המרחק המופיעים בסכומים, וכיוון שמתקיים (ראה מיד)

$$d_{1/2^k}(\vec{x}^{1/2^k}, \vec{y}^{1/2^k}) \le d(\vec{x}, \vec{y})$$

ובפרט

$$d_{1/2^k}(\vec{x}^{1/2^k}, W^{1/2^k}\vec{x}^{1/2^k}) \le d(\vec{x}, W\vec{x}) , \tag{6.25}$$

אזי החסם החדש הדוק יותר. ניתן להראות את (6.25) ע"י כתיבת הביטוי המפורש עבור המרחק, אולם ההסבר האינטואיטיבי יועיל יותר במקרה זה. נשים לב כי $\vec{x}^{1/2}$ הוא למעשה גירסה מוחלקת של $\vec{x}$. כלומר, במקום למדוד מרחק בין שני הוקטורים, אנו מודדים מרחק בין שתי גירסאות מוחלקות של האותות, ולכן אי-השיוויון שכתוב למעלה.

מספר האיברים בסכום - בעוד שבחסם הקלאסי הסכום מתבצע על אינסוף איברים, הרי שבחסם המשופר הסכום הוא על מספר איברים סופי בלבד שכפי שהראינו הם קטנים או שווים לאיברי הסכום הקלאסי (המתאימים). יתרה מזו, מספר האיברים בסכום הוא קטן, למשל עבור $B = 16$ יש בסכום 3 איברים בלבד.

לאחר שהראינו מדוע החסם המשופר הדוק יותר, נסביר את ההבדל היותר מהותי בינו ובין החסם הקלאסי. ההבדל המהותי הוא בכך שלראשונה החסם מבטא באופן מפורש תלות בתכולת התדר של האותות. כיוון שהחסם מכיל מרחקים של ורסיות מוחלקות יותר-ויותר של שגיאת הפסיפס ($\vec{x}^{1/2^k} - W^{1/2^k}\vec{x}^{1/2^k}$), הרי שהוא ייתן תוצאה שונה עבור אותות בעלי תכולת תדר שונה. כל זאת, בניגוד לחסם הקלאסי, שבו כל הביטויים עוסקים באותה רזולוציה.

עוד נקודה ראויה לציון היא העובדה שהחסם המשופר כולל בתוכו מידית את המקרה של טרנספורמציות שהן מכווצות לבסוף (eventually contractive). זאת, בניגוד לחסם הקלאסי אשר דורש מפורשות $s < 1$ (קיימת הרחבה פשוטה לחסם הקלאסי עבור טרנספורמציות מכווצות לבסוף [50], אולם היא גורמת לכך שהחסם יהיה גרוע מאד בד"כ).

6.2.1 אלגוריתם קידוד משופר

כיוון שהחסם החדש מכיל מספר קטן של איברים (בד"כ $1 - 3$), הרי שניתן לבצע מינימיזציה ישירות עליו, ולא רק על שגיאת הפסיפס. כמו כן, מכיוון שהאיברים בסכום המרכיב את החסם יורדים בערכם עם עליית k, הרי שניתן לקחת רק את שני האיברים הראשונים. כלומר, במקום שיטת הקידוד הקלאסית שהתבססה על מינימיזציה של שגיאת הפסיפס בלבד, בשיטת הקידוד החדשה אנו מציעים לבצע את המינימיזציה הבאה

$$\min_{W} \left\{ d(\vec{x}, W\vec{x}) + \alpha \cdot d_{1/2}(\vec{x}^{1/2}, W^{1/2}\vec{x}^{1/2}) \right\}. \tag{6.26}$$

כמובן, יש לזכור שלפני מציאת הקוד אין אנו יודעים את ערכו של מקדם הכיווץ s, ולכן גם אין מידע איזה ערך יש לקחת עבור α. על בעיה זו ניתן להתגבר ע"י ביצוע פעולת המינימיזציה על אוסף תמונות אימון (training set), כאשר מתיחסים אל α כאל משתנה

חופשי. לאחר מכן, לבחור את אותו גורם שהביא לתוצאות קידוד טובות ביותר, ולהשתמש בו בהמשך. אנו מצפים, לפי הביטויים בחסם, שערך זה יהיה קרוב למקדם הכיווץ הממוצע עבור הטרנספורמציות.

6.3 דוגמאות

6.3.1 דוגמא עבור אות סינטטי

בסעיף זה נדגים את חשיבות החסם החדש תוך שימוש בדוגמא פשוטה. יהיה וקטור המקור

$$\vec{x} = \begin{array}{l} [\ 60\ \ 40\ \ 24\ \ 20\ \ 24\ \ 22\ \ 20\ \ 14 \\ \ \ \ 51\ \ 49\ \ 33\ \ 27\ \ 14\ \ 8\ \ 8\ \ 2\ \ ]^T \end{array} \tag{6.27}$$

נשתמש בגודל בלוק-טווח $B = 4$ ולא נרשה חפיפה בין בלוקי התחום, כלומר ישנם 4 בלוקי-טווח ו-2 בלוקי-תחום (בעלי 8 איברים כ"א). קוד IFS מקורב עבור וקטור זה מתואר בטבלה הבאה:

הסחה b_i	בלוק-תחום מס' m_i	כיול a_i	בלוק-טווח מס' i
18	2	0.75	1
6	1	0.5	2
22	2	0.75	3
-6	1	0.5	4

$B = 4\ ,\ M_R = 4$

עבור קוד זה, נקודת-השבת היא (ואת זאת ניתן לבדוק בעזרת ניר ועיפרון),

$$\vec{f} = [\; 57 \;\; 39 \;\; 27 \;\; 21 \;\; 30 \;\; 18 \;\; 18 \;\; 14 \;\; 61 \;\; 43 \;\; 31 \;\; 25 \;\; 18 \;\; 6 \;\; 6 \;\; 2 \;]^T \qquad (6.28)$$

מקדם הכיווץ עבור הטרנספורמציה חושב ונמצא $s_1 = 0.75$ (לפי ערך-עצמי של הייצוג המטריצי; ראה סעיף 4.1.1 וגם [51]). בטבלה הבאה מוצגות תוצאות שונות עבור קוד זה ואות המקור הנתון. התוצאות מתארות את המרחק d^{l^2} (rms) בין האותות.

חסם-חדש	חסם-קלאסי	$d(\vec{x}, \vec{f})$	$d(\vec{x}, W\vec{x})$
5.1486	15.8509	3.9051	3.9627

איור 6.1: שגיאת קידוד וערך החסמים (קלאסי וחדש) עבור הדוגמא הסינטטית.

Fig. 6.1: Coding error and the Collage-boundings (classical and new) for the synthetic example.

מהתוצאות ניתן לראות שהחסם החדש הדוק הרבה יותר מהחסם הקלאסי. בעוד שהחסם הקלאסי נותן תוצאה שהיא פי 4 (!) מהערך האמיתי של שגיאת הקידוד, הרי החסם החדש נותן תוצאה שהיא פחות מ-1.5 הערך האמיתי. עוד עובדה הבולטת מהטבלה היא העובדה ששני החסמים הם עדיין רחוקים למדי מהערך האמיתי, וכמו כן ששגיאת הפסיפס ($d(\vec{x}, W\vec{x})$) קרובה מאד לשגיאת הקידוד ($d(\vec{x}, \vec{f})$).

6.3.2 ניסויים בקידוד-תמונות

בניסויים על תמונות, השתמשנו בתמונת "Lena", $256 \times 256 \times 8$bpp.

הניסוי הראשון מתאר את ערך החסם המשופר עבור קידוד התמונה "Lena". שני קודי IFS שונים נמצאו עבור תמונה זו: קוד אחד (קוד1) נעשה תוך הגבלת הטרנספורמציה להיות בעלת גורמי כיול הקטנים בערכם המוחלט מ-1, ותוך הגבלת תחום-החיפוש של בלוק-תחום לאזור ברדיוס של 96 פיקסלים מסביב לבלוק הטווח, וקוד שני (קוד2) שבו לא הוגבלו מקדמי-הכיול והמרחק בין בלוק-תחום ובלוק-טווח. בשני המקרים בלוקי-טווח הם בגודל של 8 × 8 ובלוקי-תחום בגודל 16 × 16 פיקסלים, ללא חפיפה. לא נעשה כימות של מקדמים, ולכן אין נתונים לגבי הדחיסה. המטריקה שבה השתמשנו היא d^{l^2} (rms).

חסם-חדש	חסם-קלאסי	$d(\vec{x},\vec{f})$	$d(\vec{x},W\vec{x})$	
42.30	353.00	13.11	12.66	קוד1
38.81	543.71	12.60	12.13	קוד2

איור 6.2: שגיאת קידוד וערך החסמים (קלאסי וחדש) עבור התמונה "Lena".

Fig. 6.2: Coding error and the Collage-boundings (classical and new) for image "Lena".

טבלה זו מראה מספר עובדות: א. שימוש במקודד ללא מגבלות משפר את התוצאות, במקרה שלנו ישנו שיפור של ה- PSNR ב- 0.35dB.

ב. החסם החדש הדוק יותר מהחסם הקלאסי באופן ברור. ג. עדיין קיים פער משמעותי בין החסם המשופר והערך האמיתי של שגיאת המקודד.

הניסוי השני בא לבדוק את שיטת הקידוד החדשה שהוצגה, ואשר קראנו לה קידוד עם משקלות. לא נעשו נסיונות רבים עם שיטת הקידוד החדשה שהצענו אולם, כפי שמפורט בטבלה הבאה, ביצענו מספר נסיונות קידוד עם משקלים שונים.

כפי שרואים מהטבלה, ישנו שיפור (קטן מאד) בתוצאות הקידוד. האופטימום מושג עבור

α	0	1.2	2.4	3.6	4.8
PSNR $(\vec{x}, \vec{f})$	26.85	26.91	26.93	26.91	26.88

איור 6.3: תוצאות קידוד-עם-משקלות עבור התמונה "Lena".

Fig. 6.3: Enoding-with-weights results for image "Lena".

$\alpha = 2.4$, וערך זה אינו מפתיע בהתחשב בכך שמקדם הכיווץ עבור קוד זה חושב והוא $s_1 =$ 2.36 (מקדם הכיווץ של יתר הקודים הוא מאד קרוב לערך זה). העובדה שערך מקדם הכיווץ גדול מאחד (כלומר, הטרנספורמציה אינה מכווצת) אינה מפריעה, משום שהטרנספורמציה היא מכווצת לבסוף, וכפי שכבר צוין, החסם כולל מידית גם את המקרים של טרנספורמציה מכווצת לבסוף. גם השיפור החזותי עקב השימוש בקידוד עם משקלות (שיטת הקידוד החדשה) הוא קטן, אולם הוא מרוכז רובו ככולו בחלקים ה'קשים' של התמונה, כמו למשל איזור העין.

כדי להמחיש את ההבדל, מוצגות בתמונות הבאות מספר תוצאות. בכל המקרים, גודל בלוק-טווח הוא 8, ללא חפיפה בין בלוקי-תחום.

איור 6.4: תמונת מקור: "לנה", $8 \times 256 \times 256$.

Fig. 6.4: Source image : "Lena", $256 \times 256 \times 8$.

איור 6.5: תמונה מפוענחת לאחר קידוד ללא משקלות $(\alpha = 0)$.

Fig. 6.5: Decoded image after coding without weights $(\alpha = 0)$.

איור 6.6: תוצאה לאחר קידוד-עם-משקלות, תוך שימוש ב- $\alpha = 2.4$.

Fig. 6.6: Decoded image after coding-with-weights, using $\alpha = 2.4$.

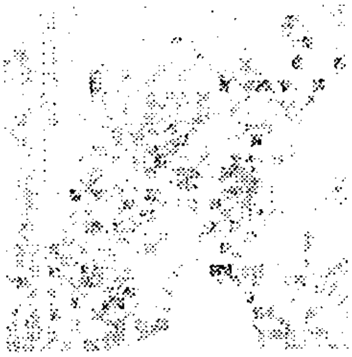
איור 6.7: תמונת הפרש בין המקור לבין פיענוח קידוד-ללא-משקלות.

Fig. 6.7: Diff. image between source and decoded images (coding without weights).

איור 6.8: תמונת הפרש בין המקור לבין פיענוח קידוד-עם-משקלות ($\alpha = 2.4$) .

Fig. 6.8: Diff. image between source and decoded images

(coding-with-weights,$\alpha = 2.4$).



איור 6.9: תמונת הפרשים בין תוצאת קידוד עם משקלות לקידוד ללא-משקלות .

Fig. 6.9: Diff. image between the decoded images (cases of coding with/without

weights $\alpha = 2.4$ .vs. $\alpha = 0$).

פרק 7

סיכום, מסקנות וכיווני המשך

7.1 סיכום ומסקנות

תחילה הוצג בעבודה זו ניסוח מקובל למקודד פרקטלי של אותות. תוצאת תהליך הקידוד הוא קוד IFS, שהוא למעשה תאור של מערכת פונקציות. מערכת פונקציות זו מתארת טרנספורמציה מכווצת, אשר נקודת השבת שלה קרובה לאות המקורי אותו קודדנו. פיענות הקוד נעשה ע"י הפעלה חוזרת של הטרנספורמציה על אות התחלה כלשהו, וכיוון שהטרנספורמציה מכווצת, מובטחת התכנסות אל נקודת השבת. לפיכך, המרחק בין נקודת השבת ובין האות המקורי הוא העיוות בקידוד.

החלק המקורי של העבודה נפתח בהצגת האפשרות של פענוח הקוד במספר רזולציות שונות. העבודה עם מספר רזולוציות שונות איפשרה להציגן במבנה פירמידלי, אשר קראנו לו פירמידת נקודות השבת. בעזרת מבנה זה הוכחנו כי קיים קשר פשוט בין הרזולציות השונות, כאשר קשר זה מובע בעזרת קוד ה-IFS. כמו כן, הוכחנו כי קיימת פונקציה (אחת ויחידה) המונחת בבסיס כל הפירמידה, וקראנו לה הפונקציה המוכלת בקוד (IFS embedded function). בעזרת פונקציה זו ניתן למצוא את נקודת השבת המתאימה לקוד בכל רזולוציה שהיא. בהמשך גם הוצג משפט אשר מלמד על המימד הפרקטלי של פונקציה זו.

את החלק הדן ביישומים פתח היישום של פענוח מהיר. בניגוד לשיטת הפענוח הרגילה, שבה מבצעים אינטרציות על אות באותה רזולוציה של האות המקורי, עד להגעה לנקודת השבת (או מספיק קרוב אליה), בשיטה החדשה מוצאים את האות ברזולוציה נמוכה, ורק לאחר מכן מגדילים את הרזולוציה עד שמגיעים לרזולוציה מלאה. החיסכון הניכר בשיטה זו נובע מכך שהאיטרציות מתבצעות על אות ברזולוציה נמוכה, בניגוד לשיטת הפענוח הרגילה.

היישום הבא שהודגם הוא לצרכי רזולוצית-על (super-resolution), או אינטרפולציה. הרעיון בבסיס יישום זה הוא להמשיך את פירמידת נקודות השבת אל מעבר לרמת הרזולוציה הנוכחית, לפי האלגוריתם שמצאנו למעבר בין רמות הפירמידה. בעזרת דוגמא הראינו שאופי האינטרפולציה הוא פרקטלי (כתלות בקוד ה-IFS כמובן), כלומר עם עליית הרזולוציה נוספים פרטים לאות, ולא מתבצעת ההחלקה המוכרת משיטות אינטרפולציה רגילות. הראינו גם כיצד ניתן לחשב, באופן ישיר (דיסקרטי, ולא ע"י אינטגרציה של הפונקציה המוכלת בקוד), כל רזולוציה המתוארת ע"י יחס רציונלי.

בחלק זה של היישומים הזכרנו גם, שאופן מציאת הקוד צריך להיות תלוי באופן שבו התקבל האות הדיסקרטי, כלומר בשיטת הדגימה. אם אין מתחשבים בשיטת הדגימה, הקוד שיתקבל עלול להיות פחות טוב מבחינת העיוות המתקבל, או מבחינת יחס הדחיסה המתקבל.

לסיום, הצגנו חסם חדש, משופר, על שגיאת הקידוד. חסם זה משתמש במידע הנלקח מתוך רזולציות שונות של האות, בניגוד לחסם הקלאסי אשר עשה שימוש רק ברזולוצית המקור. החסם החדש הודגם כהדוק באופן משמעותי יותר מהחסם הקודם, ומשמש כמוטיבציה לאלגוריתם קידוד משופר. אומנם השיפור שנמצא לא היה משמעותי, אולם הגישה

מצביעה על מספר כיווני שיפור אפשריים להמשך.

7.2 בעיות פתוחות וכיווני המשך

עבודה זו נושאת בעיקר אופי תאורטי, ולכן נשארו יישומים רבים הנגזרים ממנה שיש לבודקם באופן מעשי. כמו כן, ישנם גם מספר כיוונים תאורטיים מעניינים.

- משפט המימד הפרקטלי מתאר חסם על המימד הפרקטלי של הפונקציה המוכלת בקוד. הוא אינו מתאר כיצד ניתן לכוון את הקוד כך שיתן מימד פרקטלי מסוים, וכמו כן אינו מבטיח מהו המימד הפרקטלי המדויק (אלא רק חסם!). בהקשר זה ניתן לשפר את ההוכחה, או למצוא תנאים מסוימים, כך שהמימד הפרקטלי הנתון במשפט יהיה מדויק (ולא רק חסם). כמו כן, ניתן לנסות ולמצוא אלגוריתם, שגם אם לא יביא לקוד בעל מימד פרקטלי מסוים, ייתן הנחיות כיצד לפעול כדי להגדיל או להקטין מימד פרקטלי של קוד נתון. כלומר, בהינתן קוד שמצאנו שהמימד הפרקטלי שקשור אליו הוא D, כיצד יש לשנות את הקוד כך שמימד זה יקטן\יגדל. בעיה זו מעניינת מהסיבה הבאה: נניח שנתון מרקם כלשהוא, והמטרה היא לבצע הגדלת-רזולוציה של תמונת המרקם. אם אנו יודעים מהו המימד הפרקטלי של המרקם, הרי שנוכל להשתמש באלגוריתם לאינטרפולציה פרקטלית, כאשר הקוד שבו נשתמש יהיה בעל המימד הפרקטלי של המרקם.
- נושא בסיסי שלא טופל בעבודה זו הוא נושא ה'שרשראות' באות. 'שרשרת' ניתן להגדיר, באופן חופשי, כאוסף כל בלוקי-הטווח הקשורים ביניהם באות (יש לזכור, כמובן, שכל בלוק-תחום מורכב ממספר בלוקי-טווח). מנסיוננו בקידוד אותות, רק לעתים נדירות כל האות הוא שרשרת אחת, ובכל מקרה, ניתן באמצעים שונים לאלץ ריבוי

שרשראות. נושא השרשראות הוא חשוב, משום שהוא מתאר אזורים שונים באות. למשל, לכל שרשרת יש מימד פרקטלי משלה. לכן, ברור שבעזרת שרשראות ניתן לבצע סגמנטציה של התמונה (למשל, לאזורים המכילים קצוות ואזורים בעלי מרקמים שונים). יתרה מזו, בכל שרשרת ישנו 'גזע', שהוא אותו חלק שמושפע ומשפיע על איברים אחרים בשרשרת, וישנו חלק אשר רק מושפע מאיברי השרשרת (עלים). סביר להניח שע"י התרכזות בגזעים המתארים אות, ניתן לאפיין אותו, ואולי גם להשתמש בהם לצורכי דחיסה. שמוש נוסף אפשרי לשרשראות אלה היא סגמנטציה של האות.

- כפי שראינו, כל נקודת שבת מתקבלת מתוך הפונקציה המוכלת בקוד ע"י אינטגרציה על פני תחום מתאים. ניתן להראות שדבר זה מקביל למציאת מקדמי פרוק Haar של הפונקציה, ברזולוציה מתאימה. לפיכך, פירמידת נקודות השבת מתארת את מקדמי התמרת Haar של הפונקציה המוכלת בקוד, וקוד ה-IFS, המתאר קשר בין הרמות השונות, מתאר למעשה קשר בין מקדמי ההתמרה ברזולוציות השונות. קו מחשבה זה מוביל לשני כיוונים אפשריים. כיוון אחד הוא שימוש בבסיס אחר לפירוק האות, לאו דוקא בסיס Haar. דבר זה יחייב כמובן שינוי של פונקצית הכיווץ המרחבי בהתאם, אולם יתרונו הגדול יכול להיות בהקטנת גורם הבלוקיות בתוצאה הסופית (עקב העובדה שפונקציות הבסיס 'עוברות את גבולות הבלוקים'). כיוון שני אפשרי הוא ביצוע הקידוד על פירמידת ההפרשים. ניתן להראות שעבור אות פרקטלי, תהליך הקידוד בשני המקרים הוא זהה. היתרון לשיטה החדשה (פירמידת ההפרשים) יכול לנבוע מכך שהאות המקורי אינו פרקטלי בדיוק, ומכך שניתן לתת משקל שונה לרמות שונות בפירמידה (דבר המביא להתחשבות שונה בפסי-תדר שונים).
- עבודה עם טרנספורמציות שאינן מכווצות. כפי שכבר צוין בעבודה, המבנה הפרמידלי, והקשר בין הרמות השונות, מתקיים גם עבור טרנספורמציות שאינן מכווצות, אך

בעלות נקודת שבת. כמובן שהסרת ההגבלה על כיווציות מעשירה את אוסף הטרנספורמציות שאנו עוסקים איתן. החסרון הבולט של טרנספורמציות כאלה הוא העובדה שהן נוטות 'להגביר שגיאות', דהיינו, אם ברמה העליונה ישנה שגיאה בנקודת השבת (עקב כימות, למשל), הרי שאם נשתמש באלגוריתם ההתקדמות בין רמות הפרמידה, השגיאה (המרחק מנקודת השבת האמיתית) תגדל. לסיכום, העבודה עם טרנספורמציות שאינן מכווצות יכולה להביא לשיפור (בקידוד, באינטרפולציה, וכד'), אולם יש לשים לב לסכנות הנובעות מהיותן לא-מכווצות.

Appendix A

Direct proof of zoom-in/zoom-out theorem

In this section we will prove the following theorem:

Theorem 1 (Zoom) *Given an IFS code, it leads to* W^1 *with* $B = B_1$*, and to* $W^{\frac{1}{2}}$ *with* $B = B_1/2$*. Let the fixed-points of these transformations be* $\vec{f}^1$ *and* $\vec{f}^{\frac{1}{2}}$*, respectively, then:*

Zoom-out:

$$\vec{f}^{\frac{1}{2}}(j) = \frac{1}{2}\left\{\vec{f}^1(2j) + \vec{f}^1(2j-1)\right\}, \qquad j = (1,\cdots,\frac{N_1}{2}) \qquad \text{(A.1)}$$

where $N_1 \triangleq M_R \cdot B_1$.

Zoom-in:

$$\vec{f}^1((i-1)B_1 + j) = a_i \vec{f}^{\frac{1}{2}}((m_i - 1)D_h{}^{\frac{1}{2}} + j) + b_i \qquad \text{(A.2)}$$

$$i = (1,\cdots,M_R), \qquad j = (1,\cdots,B_1)$$

where $D_h^{\frac{1}{2}} \triangleq \frac{D_h^{1}}{2} = \frac{B_1}{2}$.

The proof is divided into two parts, according to the two parts of the theorem.

A.1 zoom-out

PROOF (Zoom-out) :

In order to prove that $\vec{f}^{\frac{1}{2}}$ given by (A.1) is indeed the fixed point of $W^{\frac{1}{2}}$, all we need to show is that it satisfies

$$\vec{f}^{\frac{1}{2}}(l) = W^{\frac{1}{2}}(\vec{f}^{\frac{1}{2}})(l), \qquad l = (1, \cdots, M_R B_1/2) \tag{A.3}$$

For the sequel, it is convenient to express l as follows:

$$l = (i-1) \cdot B_1/2 + k, \qquad i \in (1, \cdots, M_R) \quad ; \quad k \in (1, \cdots, B_1/2) \tag{A.4}$$

This expression for l emphasizes that the l'th element of $\vec{f}^{\frac{1}{2}}$ is actually the k'th element of the i'th range block of $\vec{f}^{\frac{1}{2}}$.

Also, let superscript 1 or $\frac{1}{2}$ denotes a symbol as belonging to $\vec{f}^{1}$ or $\vec{f}^{\frac{1}{2}}$, respectively. Thus, for example, $D_h^{\frac{1}{2}}$ denotes the shift between two adjacent domain-blocks in $\vec{f}^{\frac{1}{2}}$, and $D_h^{\frac{1}{2}} = B_1/2$ due to our basic assumption on parameters, namely the domain-blocks are just-touchnig.

We start by substituting $\vec{f}^{\frac{1}{2}}$ into the right-hand side of (A.3), and taking the relevant w_i

$$W^{\frac{1}{2}}(\vec{f}^{\frac{1}{2}})(l) \quad = \quad W^{\frac{1}{2}}(\vec{f}^{\frac{1}{2}})((i-1)\frac{B_1}{2} + k) \tag{A.5}$$

$$= a_i \cdot \varphi(\vec{d}^{\frac{1}{2}}_{m_i})(k) + b_i$$

substituting $\varphi(\cdot)$ for its definition

$$\begin{aligned} &= a_i \cdot \frac{1}{2}\left\{\vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+2k) + \vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+2k-1)\right\} + b_i \\ &= \frac{1}{2}\cdot\left\{[a_i\vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+2k)+b_i] + [a_i\vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+2k-1)+b_i]\right\} \end{aligned} \tag{A.6}$$

Now, let us further explore the first term in the last equation:

$$\begin{aligned} & a_i \cdot \vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+2k) + b_i = \\ &= a_i \cdot \frac{1}{2}\left\{\vec{f}^{1}((m_i-1)\cdot 2{D_h}^{\frac{1}{2}}+4k) + \vec{f}^{1}((m_i-1)\cdot 2{D_h}^{\frac{1}{2}}+4k-1)\right\} + b_i \\ &= a_i \cdot \frac{1}{2}\left\{\vec{f}^{1}((m_i-1){D_h}^{1}+4k) + \vec{f}^{1}((m_i-1){D_h}^{1}+4k-1)\right\} + b_i \\ &= a_i \cdot \varphi(\vec{d}^{1}_{m_i})(2k) + b_i \end{aligned}$$

but since $\vec{f}^1$ is the fixed-point of W^1, it follows that the last eqation is equal to

$$\vec{f}^1((i-1)B_1 + 2k) \tag{A.7}$$

Treating the second term in (A.6) the same way, leads to:

$$\begin{aligned} a_i \cdot \vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+2k-1) + b_i &= \\ \vec{f}^1((i-1)B_1 + 2k - 1) \end{aligned} \tag{A.8}$$

Thus, (A.6) can be written as:

$$\frac{1}{2}\left\{\vec{f}^1((i-1)B_1+2k) + \vec{f}^1((i-1)B_1+2k-1)\right\} \tag{A.9}$$

which, by the theorem statement (A.1), is just $\vec{f}^{\frac{1}{2}}((i-1)\frac{B_1}{2}+k)$ which, in turn, is the left side of (A.3). Q.E.D ■

A.2 zoom-in

PROOF (Zoom-in) :

The procedure here is similar to the previous one, so we will skip most of the details. It is needed to show that $\vec{f}^1$ obtained by (A.1) is a fixed point of W^1.

$$W^1(\vec{f}^1)((i-1)B_1+j) = \tag{A.10}$$

$$\begin{aligned} &= a_i \cdot \varphi(\vec{d}^1_{m_i})(j) + b_i = \\ &= a_i \cdot \frac{1}{2}\left\{\vec{f}^1((m_i-1){D_h}^1+2j) + \vec{f}^1((m_i-1){D_h}^1+2j-1)\right\} + b_i \end{aligned} \tag{A.11}$$

By (A.1) that we have just proved, the last equation reduces to

$$\begin{aligned} &= a_i \vec{f}^{\frac{1}{2}}((m_i-1)\frac{{D_h}^1}{2}+j) + b_i \\ &= a_i \vec{f}^{\frac{1}{2}}((m_i-1){D_h}^{\frac{1}{2}}+j) + b_i \\ &= \vec{f}^1((i-1)B_1+j) \end{aligned} \tag{A.12}$$

Q.E.D ■

Appendix B

Proof of IFS embedded-function

Theorem 2 (IFS Embedded Function) *Given an IFS code, there exists a unique function $G(x) \in L^\infty[0,1]$ such that a vector $\vec{v}_N \in \Re^N$ is a fixed point of the IFS iff it is equal to the function $G(x)$ at resolution $r = N$, i.e.,*

$$\vec{v}_N(j) = G_N(j), \qquad j = 1, 2, \cdots, N \tag{B.1}$$

The function $G(x)$ is called the IFS embedded function.

PROOF (IFS Embedded Function) : The proof is constructive, and is based on finding the IFS embedded function $G(x)$, and then showing that $G_N(x)$ is indeed a fixed-point of the IFS.

Suppose that the IFS is composed of M_R transformations. For each range-block $i, i = 1, \cdots, M_R$, let (a_i, b_i, m_i) denote its transformation parameters.

Look at the following analogies between the current case of continuous variable and the previously discussed discrete case (vectors):

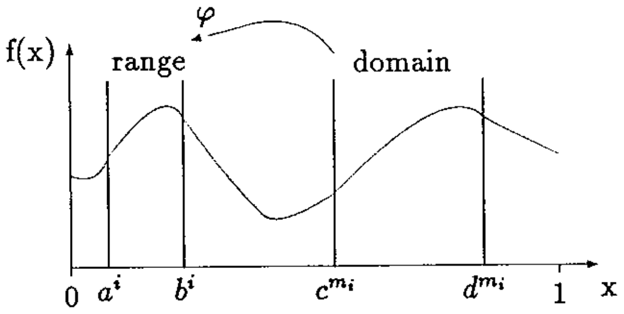


Figure B.1: Domain to range transformation

discrete	$\vec{v}_N(i)$	$i \in (1, 2, \cdots N)$	B	$D = 2B$	$D_h = B$
continuous	$f(x)$	$x \in [0, 1]$	$\frac{B}{r}$	$2 \cdot \frac{B}{r}$	$\frac{B}{r}$

discrete	i'th range-block : $\vec{v}_N(j)$,	$j = (i-1)B + 1, \cdots, iB$
continuous	i'th range-block : $f(x)$,	$x \in ((i-1)\frac{B}{r}, \frac{B}{r})$

Such a configuration is demonstrated in Fig. B.1, where the i'th range-block is

$$x \in (a^i, b^i), \qquad a^i = (i-1)\frac{B}{r} \qquad \text{and} \qquad b^i = i\frac{B}{r} \tag{B.2}$$

and the m_i'th domain-block is

$$x \in (c^{m_i}, d^{m_i}), \qquad c^{m_i} = (m_i - 1)\frac{B}{r} \qquad \text{and} \qquad d^{m_i} = (m_i - 1)\frac{B}{r} + 2 \cdot \frac{B}{r} \tag{B.3}$$

Define the following transformation $w_i : L^\infty[0,1] \to L^\infty[0,1]$:

$$w_i(G(x)) = \begin{cases} a_i G(2(x - a^i) + c^{m_i}) + b_i & a^i \leq x < b^i \\ 0 & otherwise \end{cases} \tag{B.4}$$

Then, since $|a_i| < 1$, w_i is contractive. Next, define the transformation:

$$W = \bigcup_{i=1}^{M_R} w_i \tag{B.5}$$

which is a contractive IFS [32], and thus has a unique fixed point, defined to be the function $G(x)$.

Now it is left to show that $G_N(x)$ is indeed a fixed-point of the IFS code. Take

$$\vec{v}_N(i) = G_N(i) \tag{B.6}$$

and look for consistency with the fixed point criterion, i.e. verify that $\vec{v}_N = W(\vec{v}_N)$. Take for example, the j'th element in the i'th range-block. Let (a_i, b_i, m_i) denote the appropriate transformation parameters, so the fixed-point equation for this element is:

$$\begin{aligned} \vec{v}_N((i-1)B+j) &= a_i\frac{1}{2}(\vec{v}_N((m_i-1)B+2(j-1))+ \\ &\quad \vec{v}_N((m_i-1)B+2(j-1)+1))+b_i \qquad \text{(B.7)} \\ &\quad j=1,2,\cdots,B \end{aligned}$$

This time we deal with elements rather then whole blocks, so that the following notations are used (see also Fig. B.2):

- Boundaries of the j'th element in the i'th range-block are

$$\tilde{a} = ((i-1)B+j)\frac{1}{r} \qquad \text{and} \qquad \tilde{b} = ((i-1)B+j+1)\frac{1}{r} \tag{B.8}$$

- Boundaries of the j'th and the (j+1)'th elements in the m_i domain-block are

$$\tilde{c} = ((m_i-1)B+2(j-1))\frac{1}{r}, \qquad \tilde{e} = \frac{\tilde{c}+\tilde{d}}{2} = ((m_i-1)B+2(j-1)+1)\frac{1}{r} \tag{B.9}$$

$$\tilde{d} = ((m_i-1)B+2(j-1)+2)\frac{1}{r}. \tag{B.10}$$

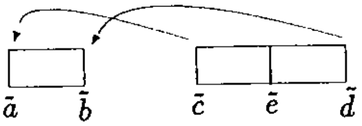


Figure B.2: Elements of transformations

Substituting $G_N(i)$ for $\vec{v}_N(i)$ in the right side of (B.7) yields:

$$\begin{aligned}\vec{v}_N((i-1)B+j) &= a_i\frac{1}{2}r(\int_{\tilde{c}}^{\tilde{e}} G(x)\,dx + \int_{\tilde{e}}^{\tilde{d}} G(x)\,dx) + b_i \\ &= a_i\frac{1}{2}r\int_{\tilde{c}}^{\tilde{d}} G(x)\,dx + b_i \qquad \text{(B.11)}\end{aligned}$$

The left side of (B.7) yields

$$\vec{v}_N((i-1)B+j) = r\int_{\tilde{a}}^{\tilde{b}} G(x)\,dx \qquad \text{(B.12)}$$

since $G(x) = w_i(G(x))$, and remembering that the range $x \in [\tilde{c}, \tilde{d}]$ is mapped to $x \in [\tilde{a}, \tilde{b}]$,

$$\begin{aligned}& r\int_{\tilde{a}}^{\tilde{b}}[a_i G(2(x-\tilde{a})+\tilde{c})+b_i]\,dx \\ =\ & a_i r\int_{\tilde{a}}^{\tilde{b}} G(2(x-\tilde{a})+\tilde{c})\,dx + r\int_{\tilde{a}}^{\tilde{b}} b_i\,dx \\ =\ & a_i\frac{1}{2}r\int_{\tilde{c}}^{\tilde{d}} G(x)\,dx + b_i \qquad \text{(B.13)}\end{aligned}$$

Thus, since (B.11) and (B.13) agree, we've established the consistency of the fixed-point equation for $\vec{v}_N$, under the theorem assumption that $\vec{v}_N(i) = G_N(i)$.

The uniqueness of the embedded-function also follows immediately from the fact that $\vec{v}_N(i) = G_N(i) \quad \forall N$.

Q.E.D ▪

Appendix C

Proof of zoom-in/zoom-out theorem using IFS embedded function

Before getting into the details of the proof, it is worth noting the difference between the proof that follows, and that given in Appendix A. As implied by the title, the following proof is based on the embedded-function theorem. This is unlike the proof in Appendix A. The fact that the proof relies on the embedded-function theorem, makes the proof apllicable only under the condition that the IFS is contractive. This is unlike the previous proof, which holds also for *non-contractive* IFS. The only condition there is that there is a fixed-point for the IFS at some resolution.

Now, the theorem will be restated, and than proved.

Theorem 3 (Zoom) *Given an IFS code, it leads to* W^1 *with* $B = B_1$*, and to* $W^{\frac{1}{2}}$ *with* $B = B_1/2$*. Let the fixed-points of these transformations be* $\vec{f}^1$ *and* $\vec{f}^{\frac{1}{2}}$*, respectively, then:*

Zoom-out:

$$\vec{f}^{\frac{1}{2}}(j) = \frac{1}{2}\left\{\vec{f}^1(2j) + \vec{f}^1(2j-1)\right\}, \qquad j = (1, \cdots, \frac{N_1}{2}) \tag{C.1}$$

where $N_1 \triangleq M_R \cdot B_1$.

Zoom-in:

$$\vec{f}^1((i-1)B_1 + j) = a_i \vec{f}^{\frac{1}{2}}((m_i - 1){D_h}^{\frac{1}{2}} + j) + b_i \tag{C.2}$$

$$i = (1, \cdots, M_R), \qquad j = (1, \cdots, B_1)$$

where ${D_h}^{\frac{1}{2}} \triangleq \frac{{D_h}^1}{2} = \frac{B_1}{2}$.

The proof is divided into two parts, according to the two parts of the theorem.

C.1 zoom-out

PROOF (Zoom-out) :

According to the embedded-function theorem (Appendix B.), all one need to show in order to prove that $\vec{f}^{\frac{1}{2}}$ is a fixed-point of the givrn IFS, is:

$$\vec{f}^{\frac{1}{2}}(j) = G_{N/2}(j) = \frac{N}{2}\int_{(j-1)/(N/2)}^{j/(N/2)} G(x)\, dx. \tag{C.3}$$

where $G(x)$ is the IFS embedded-function.

To that end, we will substitute the appropiate terms for the rhs of equation (C.1),

$$\begin{aligned}\frac{1}{2}\left\{\vec{f}^{1}(2j)+\vec{f}^{1}(2j-1)\right\} &= \frac{N}{2}\cdot\left\{\int_{(2j-1)/N}^{2j/N} G(x)\,dx+\int_{(2j-2)/N}^{2j-1/N} G(x)\,dx\right\}\\ &= \frac{N}{2}\int_{(2j-2)/N}^{2j/N} G(x)\,dx\\ &= \frac{N}{2}\int_{(j-1)/(N/2)}^{j/(N/2)} G(x)\,dx\end{aligned}$$

and thus we have proved the theorem. Q.E.D ■

C.2 zoom-in

The proof is very similar to the previous one, when using the same notations as in Appendix B, namely $\tilde{a}$,$\tilde{b}$,$\tilde{c}$,$\tilde{d}$,$\tilde{e}$. Therefore, the proof will be omitted.

Appendix D

Proof of fractal dimension of IFS-embedded-function

Theorem 4 (Fractal Dimension of IFS embedded function) *Given an IFS code W, let $[F], [A], [D]$, and $\vec{b}$ denote its matrix representation using $B = 1$, such that*

$$W(\vec{v}) = [F]\vec{v} + \vec{b} = ([A]\frac{1}{2}[D])\vec{v} + \vec{b} \qquad \text{(D.1)}$$

Let $G(x)$ denote the related IFS embedded function. The fractal dimension of $G(x)$ is

$$1 \leq \mathcal{D} \leq 1 + \log_2(\lambda)) \qquad \text{(D.2)}$$

where λ is the largest real eigen-value of the matrix $([|A|][D])$, and $[|A|]$ denotes the absolute value of $[A]$.

PROOF (Fractal Dimension of IFS Embedded Function) : A related formal proof can be found in [3, 6]. We will follow the informal proof, similar to the one given in [4] about the fractal dimension of a fractal interpolation function.

Let $\epsilon > 0$, and let $G(x)$ be superimposed on a grid of square-boxes of side length ϵ. Let $N(\epsilon)$ denote the number of grid boxes of side length ϵ which intersect $G(x)$. Since $G(x)$ has a fractal dimension $\mathcal{D}$, it follows that

$$N(\epsilon) \sim const \cdot \epsilon^{-\mathcal{D}} \qquad as \qquad \epsilon \to 0 \tag{D.3}$$

Our goal is to evaluate the value of $\mathcal{D}$.

Denote the number of boxes intersecting $G(x)$ when $x \in ((i-1)\frac{1}{r}, i\frac{1}{r})$ by

$$N_i(\epsilon), \qquad i = (1, \cdots, M_R) \tag{D.4}$$

Namely, $N_i(\epsilon)$ is the number of intersecting boxes in the i'th range-block.

Since $\epsilon \to 0$, the contribution of boxes at the edges of range-blocks can be ignored, therefore we can write

$$N(\epsilon) \cong \sum_{i=1}^{M_R} N_i(\epsilon) \tag{D.5}$$

Now, let us investigate more thoroughly the value of $N_i(\epsilon)$. Let m_i denote the index of the domain-block mapped to the *i'th* range-block, according to the IFS. This domain block is composed of two adjacent range-blocks, with indices denoted as m_i^1 and m_i^2.

After transforming the m_i'th domain-block onto the i'th range-block, each column of grid-blocks is mapped into a column of grid-parallelograms, with height $|a_i|\epsilon$ and width $\frac{B}{D}\epsilon$. Let us define $q \triangleq \frac{B}{D}$, and according to our assumptions, $q = \frac{1}{2}$. Therefore, if a column of width ϵ on the domain-block intersects $G(x)$ in L boxes, then after transformation the column will be of width $q\epsilon$, and the number of boxes of size $q\epsilon$ intersecting $G(x)$ in the transformed column is therefore bounded by $|a_i|\, L/q$ boxes.

Summing on all the domain columns yields

$$N_i(q\epsilon) \leq \frac{|a_i|}{q}(N_{m_i^1}(\epsilon) + N_{m_i^2}(\epsilon)) \tag{D.6}$$

Denoting by c_i the proportion factor

$$N_i(\epsilon) \sim c_i \cdot \epsilon^{-\mathcal{D}} \qquad as \qquad \epsilon \longrightarrow 0 \tag{D.7}$$

we can write

$$c_i \cdot (q\epsilon)^{-\mathcal{D}} \leq \frac{|a_i|}{q}(c_{m_i^1}\epsilon^{-\mathcal{D}} + c_{m_i^2}\epsilon^{-\mathcal{D}}) \tag{D.8}$$

The same reasoning can be argued for any range-block, to yield

$$c_i \cdot (q)^{-\mathcal{D}+1} \leq \ |a_i|\,(c_{m_i^1} + c_{m_i^2}), \qquad i = (1, \cdots, M_R) \tag{D.9}$$

Recalling the definition of the matrix $[D]$, the last equation, for all the blocks, can be written as

$$\vec{c} \cdot (q)^{-\mathcal{D}+1} \leq (\ |[A]|\ \cdot [D]) \cdot \vec{c}, \qquad \vec{c} = [c_1, c_2, \cdots, c_{M_R}]^T. \tag{D.10}$$

where the last vector inequality should be understood term-wise.

By the last inequality, and since c_i is a non-negative vector, and $([|A|] \cdot [D])$ is a non-negative matrix, $(q)^{-\mathcal{D}+1}$ is bounded from above by the largest real eigen-value of $([|A|] \cdot [D])$ (see [63], exercise 1.12, p.25).

If λ denotes the largest real eigen-value, then

$$\begin{aligned} (q)^{-\mathcal{D}+1} &\leq \lambda \\ \mathcal{D} &\leq 1 + \log_{\frac{1}{q}}(\lambda) \end{aligned} \tag{D.11}$$

Since we assume $q = \frac{B}{D} = \frac{1}{2}$, the theorem follows immediately. Q.E.D ■

It is worth noting that:

- Equation (D.10) holds even when $q \neq \frac{1}{2}$, and so does (D.16). For example, it can be shown immediately that in the case of uniform fractal-interpolation of functions [4], where the whole function is mapped to each interval, the following results:

$$D = N, \qquad q = \frac{D}{B} = M_R \tag{D.12}$$

$$1 \leq \mathcal{D} \leq 1 + \log_{M_R}\left(\sum_{i=1}^{M_R} |a_i|\right) \tag{D.13}$$

  which agrees with the result there.

- The matrix $q \cdot [D]$ is a stochastic matrix, i.e. the sum of the elements in each row equals 1. Thus, all its eigen-values are equal or smaller then 1 (in magnitude), and at least it has one eigen-value equal to 1. For example, if

$$[A] = a \cdot [I]_{M_R \times M_R} \tag{D.14}$$

  one can write

$$|[A]| \cdot [D] = |a| \, [I]_{M_R \times M_R} \cdot \frac{1}{q} q[D] = \frac{|a|}{q} \cdot (q[D]) \tag{D.15}$$

  Since $|a|/q$ is a scalar, and $q[D]$ is stochastic, and therfore the largest eigen-value is $|a|/q$. Thus, (D.16) turns to

$$1 \leq \mathcal{D} \leq 2 - \log_{\frac{1}{q}}(|a|) \tag{D.16}$$

Appendix E

Super-resolution Pictures

This appendix describes the results of the fractal based super-resolution technique described in the work.

In the following, the source picture was taken from the 'Table Tennis' scenario, first frame. Only part of it of size 168 × 112 (8 bits/pixel) is considered (see Fig. E.1). This picture is then 'shrinked', by a factor of 2 in each direction, by a process of averaging followed by decimation.

The smaller picture is then 'zoomed-in' to the original size, in order to compare different methods of super-resolution:

1. Bilinear interpolation, which is a simple 2-D linear interpolation (Fig. E.4).
2. Fractal interpolation, based on coding the 'small' image with range-block size of $B = 2$ *(no rotations)* and then decoding with $B = 4$ (Fig. E.5). The $B = 4$ level can be found directly via 'zooming' on the $B = 2$ level.

When looking at the figures, it is well noted that the fractal interpolation technique suits best the textured areas, whereas its performance in areas containing edges is quite poor. This suggests the use of this technique only for textured areas.

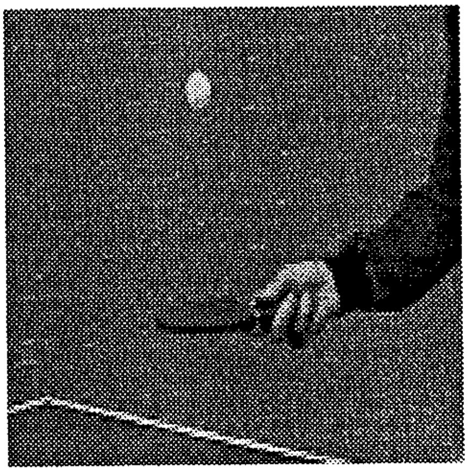


Figure E.1: Original picture.

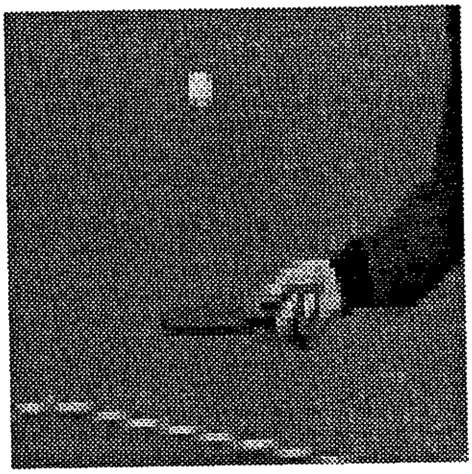


Figure E.2: 'Smaller' picture (blown-up by enlarging pixels).

Figure E.3: Result of coding 'smaller' picture, using $B = 2$ and no rotations (blown-up by enlarging pixels).

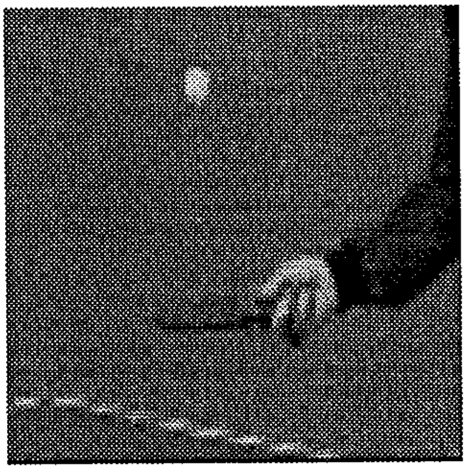


Figure E.4: Interpolated picture using Bilinear interpolation.

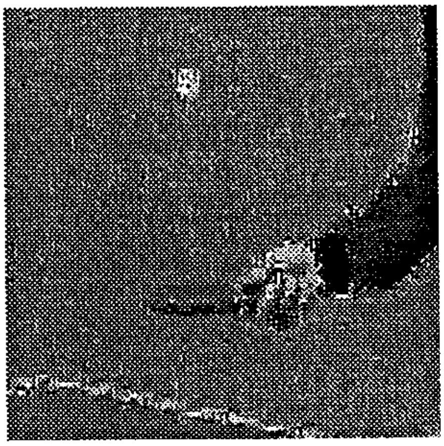


Figure E.5: Interpolated picture using Fractal-method.

מקורות

[1] Fritz Albregtsen et al. Fractal dimension, only a fraction of the truth? pages 733–736. ICPR-11, 1992.

[2] Z. Baharav, D. Malah, and E. D. Karnin. Hierarchical interpretation of fractal image coding and its application to fast decoding. pages 190–195, Cyprus, July 1993. Intl. Conf. on Digital Signal Processing.

[3] M. F. Barnsley. Fractal functions and interpolation. *Constr. Approx.*, 2:303–329, 1986.

[4] M. F. Barnsley. *Fractals Everywhere.* Academic-Press, New York, 1988.

[5] M. F. Barnsley and S. Demko. Iterated function systems and the global construction of fractals. *Proc. of Royal Soc. London*, A399:243–275, 1985.

[6] M. F. Barnsley, J. H. Elton, and D. P. Hardin. Recurrent iterated function systems. *Constructive approx.*, 1:3–31, 1989.

[7] M. F. Barnsley et al. Harnessing chaos for image synthesis. *Computer Graphics*, 22(4):131–140, 1988.

[8] M. F. Barnsley and L. P. Hurd. *Fractal Image Compression.* AK Peters, Massachusetts, 1993.

[9] M. F. Barnsley and A. E. Jacquin. Application of recurrent iterated function systems to images. volume 1001, pages 122–131. SPIE-Visual Communications and Image Processing, 1988.

[10] M. F. Barnsley and A. D. Sloan. A better way to compress images. *Byte*, pages 215–223, January 1988.

[11] K. U. Barthel, T. Voye, and P. Noll. Improved fractal image coding. Lausanne-Switzerland, March 1993. PCS-Picture Coding Symposium.

[12] J. M. Beaumont. Image data compression using fractal technique. *BT Technol. J.*, 9(4):93–109, October 1991.

[13] E. L. J. Bohez et al. Fractal dimension and iterated function system (ifs) for speech recognition. *Electronics Letters*, 28(15):1382–1384, July 1992.

[14] Peter J. Burt and Edward H. Adelson. The laplacian pyramid as a compact image code. *IEEE Trans. on communications*, 31(4):532–540, April 1983.

[15] S. Chen and J. M. Keller. Texture description and segmentation through fractal geometry. *Computer Vision, Graphics, and Image Processing*, 45:150–166, 1989.

[16] Y. Cheng and D. M. Etter. Analysis of an adaptive technique for modeling sparse systems. *IEEE Trans. on ASSP*, 37(2):254–264, February 1989.

[17] Harvey A. Cohen. Deterministic scanning and hybrid algorithms for fast decoding of ifs encoded image sets. volume 3, pages 509–512. ICASSP, 1992.

[18] Robert L. Devaney. *An introduction to Chaotic Dynamical Systems.* Adison-Wesley, 1989.

[19] Kenneth Falconer. *Fractal Geometry,mathematical foundations and applications.* Wiley and Sons, New-York, 1990.

[20] Y. Fisher, E. W. Jacobs, and R. D. Boss. *Image and Text Compression*, chapter :Fractal Image Compression Using Iterated Transforms, pages 35–61. Kluwer Academic Publishers, 1992.

[21] Y. Fisher, E. W. Jacobs, and R. D. Boss. Image processing: A study of iterated transform method. *Signal Processing*, 29(3):251–263, December 1992.

[22] Yuval Fisher. *Fractal Compression: Theory and applications to Digital Images.* Springer Verlag, 1994.

[23] Q. Gan, K. Kotani, and M. Miyahara. Quantization of ulcs color space using the peano scan. Lausanne-Switzerland, March 1993. PCS-Picture Coding Symposium.

[24] Allen Gersho and Robert M. Gray. *Vector Quantization and Signal Compression.* Kluwer Academic Publishers, 1992.

[25] M. Gharavi-Alkhansari and T. S. Huang. A fractal-based image block-coding algorithm. Lausanne-Switzerland, March 1993. PCS-Picture Coding Symposium.

[26] M. Gharavi-Alkhansari and T. S. Huang. A fractal-based image block-coding algorithm. volume 5, pages 345–348. ICASSP, 1993.

[27] D. P. Hardin and P. R. Massopust. The capacity for a class of fractal functions. *Communications in mathematical physics*, 105:455–460, 1986.

[28] D. A. Hoskins and J. Vagners. Image compression using iterated function systems and evolutionary programming: Image compression without image metrics. volume 2, pages 705–711. 26th Asilomar Conf. on Signals, Systems and Computers, 1992.

[29] Bernd Hürtgen, F. Müller, and C. Stiller. Adaptive fractal coding of still pictures. Lausanne-Switzerland, March 1993. PCS-Picture Coding Symposium.

[30] K. Culik II, S. Dube, and P. Rajcani. Efficient compression of wavelet coefficients for smooth and fractal-like data. Snowbird-Utah, April 1993. 1993 Data Compression Conference, IEEE Computer Society Press.

[31] A. E. Jacquin. Fractal image coding: A review. *Pro. of the IEEE*, 81(10):243–275, October 1993.

[32] Arnaud E. Jacquin. *A Fractal Theory of Iterated Markov Operators with applications to Digital Image Coding.* PhD thesis, Georgia Institute of Technology, August 1989.

[33] Arnaud E. Jacquin. Image coding based on a fractal theory of iterated contractive image transformations. *IEEE Transactions on Image Processing*, 1(1):18–30, January 1992.

[34] Steve Kocsis. Digital compression and iterated function systems. volume 1153, pages 19–27. SPIE-Application of Digital Image Processing, 1989.

[35] W. G. Kropatch et al. Combining pyramidal and fractal image coding. pages 61–64. ICPR-11, 1992.

[36] Skjalg Lepsøy. *Attractor Image Compression: Fast Algorithms and Comparisons to Related Techniques.* PhD thesis, The Norwegian Inst. of Technology, June 1993.

[37] Skjalg Lepsøy et al. Attractor image compression with a fast non-iterative decoding algorithm. volume 5, pages 337–340. ICASSP, 1993.

[38] Lars Lundheim. *Fractal Signal Modelling for Source Coding.* PhD thesis, The Norwegian Inst. of Technology, November 1992.

[39] F. J. Malassenet. Wavelet representations and coding of self-affine signals. pages 677–680. ICASSP, 1991.

[40] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on PAMI*, 11(7):674–693, July 1989.

[41] Benoit B. Mandelbrot. *Fractals form,chance,and dimension.* W. H. Freeman and Company, New York, 1977.

[42] Benoit B. Mandelbrot. *The Fractal Geometry of Nature.* W. H. Freedman and Company, New York, 1982.

[43] D. S. Mazel and M. H. Hayes. Using iterated function systems to model discrete sequences. *IEEE Trans. on Signal Processing*, 40(7):1724–1734, July 1992.

[44] P. Meer, E. S. Baugher, and A. Rosenfeld. Frequency domain analysis and synthesis of image pyramid generating kernels. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9(4):512–522, 1987.

[45] D. M. Monro. Generalized fractal transforms: Complexity issues. Snowbird-Utah, April 1993. DCC-Data Compression Conference, IEEE Computer Society Press.

[46] D. M. Monro and F. Dudbridge. Fractal approximation of image blocks. volume 3, pages 485–488. ICASSP, 1992.

[47] Mirek Novak. Attractor coding of images. Lausanne-Switzerland, March 1993. PCS-Picture Coding Symposium.

[48] Geir E. Øien et al. An inner product space approach to image coding by contractive transformations. pages 2773–2776. ICASSP, 1991.

[49] Geir E. Øien et al. Reducing the complexity of a fractal-based image coder. pages 1353–1356, 1992.

[50] Geir O. Øien. *Attractor Coding of Images.* PhD thesis, The Norwegian Inst. of Technology, June 1993.

[51] Geir O. Øien, Zachi Baharav, and Others. A new improved collage theorem with applications to multiresolution fractal image coding. In *ICASSP*, 1994.

[52] Peitgen, Jurgens, and Saupe. *Fractals fot the Classroom*, volume 1. Springer-Verlag, 1991.

[53] Heinz-Otto Peitgen and Dietmar Saupe. *The Science of Fractal Images.* Springer-Verlag, New-York, 1988.

[54] Shmuel Peleg et al. Multiple resolution texture analysis and classification. *PAMI*, 6(4):518–523, July 1984.

[55] A. Pentland and B. Horowitz. A practical approach to fractal-based image compression. volume 1605, pages 176–185. SPIE, 1991.

[56] Alex P. Pentland. Fractal-based description of natural scenes. *PAMI*, 6(6):661–674, November 1984.

[57] H. Raittinen and K. Kaski. Fractal based image compression with affine transformations. Snowbird-Utah, April 1993. DCC-Data Compression Conference, IEEE Computer Society Press.

[58] R. Rinaldo and A. Zakhor. Fractal approximation of images. Snowbird-Utah, April 1993. 1993 Data Compression Conference, IEEE Computer Society Press.

[59] A. Rosenfeld, editor. *Multiresolution Image Processing and Analysis.* Springer-Verlag, 1984.

[60] H. L. Royden. *Real Analysis.* Macmillan Pub., 1988.

[61] Walter Rudin. *Principles of Mathematical Analysis.* McGraw-Hill, 3rd edition, 1976.

[62] J. Samarabandu, R. Achyara, E. Hausmann, and K. Allen. Analysis of bone x-rays using morphological fractals. volume 3, pages 133–136. ICASSP, 1992.

[63] E. Seneta. *Non-Negative Matrices.* John Wiley and Sons, New York, 1973.

[64] T. R. Senevirathne et al. Amplitude scale method: New and efficient approach to measure fractal dimension of speech waveforms. *Electronics Letters*, 28(4):420–422, February 1992.

[65] Jaroslav Stark. Iterated function systems as neural networks. *Neural Networks*, 4:679–690, 1991.

[66] M. C. Stein and K. D. Hartt. Nonparametric estimation of fractal dimension. *SPIE-Visual Communications and Image Processing*, 1001:132–137, 1988.

[67] M. Temerinac et al. An efficeient image compression algorithm based on filter bank analysis and fractal theory. *Signal Processing*, pages 1373–1376, 1992.

[68] L. Thomas and F. Deravi. Pruning of the transform space in block-based fractal image compression. pages 341–344. ICASSP, 1993.

[69] K. S. Thyagarajan and Shankar Chatterjee. Fractal scanning for image compression. volume 1, pages 467–471. Asilomar conf. on Signals,Systems and Computers, 1991.

[70] G. Vines and M. H. Hayes. Adaptive ifs image coding with proximity maps. volume 5, pages 349–352. ICASSP, 1993.

[71] E. Walach and E. Karnin. A fractal based approach to image compression. pages 529–531. ICASSP, 1986.

[72] David S. Watkins. *Fundamentals of Matrix Computations.* Wiley and Sons, New-York, 1991.

Hierarchical Approach for Fractal Representation of Signals

Research Thesis

Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science

in

ELECTRICAL ENGINEERING

BY

ZACHI BAHARAV

Submitted to the Senate of the Technion-Israel Institute of Technology

Adar 5754 Haifa February 1994

The work described herein was supervised by Prof. David Malah and Dr. Ehud Karnin at the Faculty of Electrical Engineering - Technion, Haifa.

Dr. Ehud Karnin is with I.B.M. Israel.

I would like to thank *Prof. David Malah* and *Dr. Ehud Karnin* for their supervision.

I also thank the Signal Processing Lab. personal, and especially *Ziva Avni*, for their technical support.

I would also like to thank all those whom I held fruitfull discussions with, especially *Dr. Geir O. Øien* from Norway and *Dr. Yuval Fisher* from Los-Angles .

Lastly, I would like to thank the examiners, and especially Prof. Marc Berger, for their constructive suggestions and comments.

To my parents, Aviva and Jacob Baharav

Contents

Contents (continued)

Contents (continued)

List of Figures

List of Figures (continued)

ABSTRACT

At the begining of the work, the basics of a block oriented fractal image coder are reviewed. The output of the coder is an IFS (Iterated Function System), which approximates the image as a fixed point of a contractive transformation. The original contributions of the work starts with the introduction of a new hierarchical interpretation of the IFS code, which relates the different scales of the fixed point. We prove the existence of a unique function of a continuous variable that is associated with the IFS code. It is further shown that the different scales of the IFS fixed point are directly computable from this so called *IFS embedded function.*

Preceding the applications part of the work, the matrix representation of the IFS code is described. An example is given, where the contraction factor of a transformation is computed via its matrix representation.

The applications part begins with the use of the hierarchical representation for fast decoding, leading typically to an order of magnitude reduction of the computation time. This fast decoding procedure applies an iterative scheme for finding the signal at low-resolution, and then a deterministic algorithm for advancing to the original resolution. Another application is a new super-resolution method based on the IFS-code of a signal. The super-resolution technique is demonstrated, and its characteristics are analyzed. In a new theorem, a (tight) bound on the fractal-dimension of the IFS embedded-function is given. This fractal-dimension also tells us about the appearence of the super-resolution signal. To conclude this part of applications, the relation between the sampling method of the signal, and the IFS-coding method, is described.

In a seperate part, a new improved bound on the coding error is given. The new bound takes into account the different scales of the signal. Its advantage over the previously known bound is twofold: it holds for eventually contractive transformations, and it is a much tighter bound. This bound suggests a novel encoding scheme, which is also demonstrated.

Finally, conclusions are given and several future-directions are proposed.