

מבט אחר על פילטרים

אריה האס – 4X6FT

נובמבר 05

1. קצת היסטוריה

בתחילת שנות השמונים פרסמה האגודה ספר ובו מידע חשוב ושימושי לחובב. בין היתר, היה בספר מאמר על פילטר שתכנתי ובניתי. הפילטר תוכנן על פי טבלאות שחישבתי בעמל רב לפי שיטה שפורסמה באחד הפרסומים המקצועיים.

כשנתיים אחר כך- כבר היה באפשרותי לחשב את הפרמטרים של הפילטר בעזרת מחשב אישי. שנה אחת נוספת חלפה וכבר היתה תוכנית מחשב (לא שלי) שבחרה את סוג הפילטר וחישבה את כל רכיביו על פי נתונים או דרישות בדיאלוג פרטני עם המחשב.

חלפו להן עוד שנתיים והמחשב עצמו הפך להיות הפילטר. לא ממש בזמן אמת (מקבל הקלטות ומעבד אותן) אבל באיכות ללא תחרות ובמגוון עצום של תכונות.

נקודת המפנה מבחינתי היתה באותו יום שבו הותקנה במחשב שלי הגירסה הראשונה של MATLAB. פילטר הפסיק להיות מעגל בנוי מרכיבים אלקטרוניים אלא מערכת של ישויות מתמטיות הפועלות יחד כדי לקיים פונקציית תמסורת יחודית הניתנת לשליטתח המלאה.

בפרקים הבאים אסקור כמה טכניקות לעיבוד אות שיש בהן עניין לחובב. לבסוף אראה כמה שימושים לתחנה ולבקרה.

לכל ההדגמות להלן אעשה שימוש בחבילת תוכנה 'חינמית' לחישובים מתמטיים בשם OCTAVE. התוכנה זמינה באינטרנט עבור מערכת הפעלה Windows וגם עבור Linux. שפת התכנות היא 'עילית' מאוד וה- syntax כמעט זהה ל- MATLAB. ה- scripts המגדירים את החישובים השונים יופיעו כאן בכתב נטוי ותפוקות המחשב בכתב ישר.

2. הגדרות בסיסיות

א. כל אות שהוא (למשל - מוצא שמע של מקלט) מכיל מידע ורעש. שניהם נחשבים לאקראיים לפני שהאות עבר עיבוד כלשהוא. כלומר - אין לנו מידע מוקדם לגבי תכולת המידע והרכב הרעש.

ב. לכל אות המוצג לפנינו - ידוע לנו שיש מידע חבוי באות ואנחנו רוצים להפיק אותו בצורה המיטבית.

ג. מחשב 'מבין' אות ע"י דגימה מחזורית שלו בקצב התלוי ברוחב הסרט הנדרש. תיאורטית - קצב הדגימה של אות חייב להיות לפחות פי שניים מהתדר הגבוה ביותר שאותו רוצים להפיק מהאות. עיקרון זה נקרא 'קריטריון Nyquist' על שם המתמטיקאי שהגה אותו. מעשית - קצב הדגימה צריך להיות הרבה יותר גבוה כדי לקבל ביצועים סבירים. השפעת קצב הדגימה תובהר בהמשך.

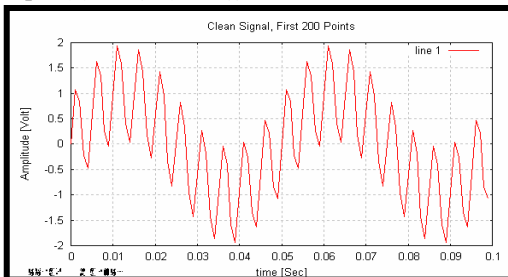
ד. אות המבחן שלנו מורכב מ- 1024 דגימות בשנייה אחת והוא כולל שני תדרים: 20 ו- 200 הרץ. כמו כן- הוא כולל רעש אקראי.

הטקסט להלן מייצר את הדגימות של האות הנקי (לפני הוספת הרעש).

```
t=0:0.001:1.024;  
w1=2*pi*20;w2=2*pi*200;  
s=sin(w1*t)+sin(w2*t);
```

אפשר לראות את 100 הנקודות הראשונות של האות בצורה גרפית ע"י הפקודות:

```
title("Clean Signal, First 200 Points");  
xlabel("time [Sec]");ylabel("Amplitude [Volt]");  
plot(t(1:100),s(1:100));
```



איור 1 : 200 הנקודות הראשונות לפני הוספת הרעש

הגרף המתקבל מופיע באיור 1 להלן:

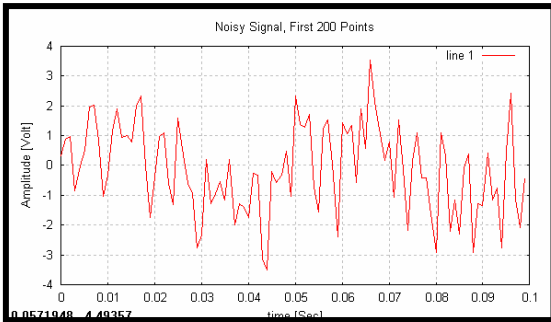
לכאורה - עדין ניתן להבחין בכיורר שישנם שני תדרים ברורים וניתן בקלות להפריד ביניהם בעזרת פילטרים קונוונציונליים.

נבנה רעש אקראי בעל פילוג נורמלי פילוג נורמלי מגדיר את התכונות הסטטיסטיות של הרעש. ונוסיף אותו לאות המקור בעזרת הפקודות:

```
noise=randn(1,length(t));  
signal=s+noise;
```

יש לחדש את הגרף שנית ע"י:

```
title("Noisy Signal, First 200 Points");
plot(t(1:100),signal(1:100));
```



איור 2 : האות המקורי בתוספת רעש

ראה איור 2 להלן.

באיור המראה את האות עם הרעש כבר קשה יותר להבחין בתכולה של האות. בטבע זה יהיה קשה הרבה יותר מאחר ולאות יהיו מספר תדרים שחלקם יראו כמו הרעש.

3. אסטרטגיות לסינון האות.

3.1 ממוצעים חלקיים

בשיטת הממוצעים החלקיים, מקיבים חלק מהידע או מהדגימות של האות לטובת 'ניקיון' האות.

מחלקים את כל הבלוק בן 1024 הנקודות ל- n חלקים שווים ומחשבים את הממוצע של כל חלק כגודל המייצג אותו במרכזו. מובן שיש לבחור את n כך שלא כל המידע באות יעלם לאחר הסינון. למשל אם n=1 אזי מקבלים ממוצע אחד לכל בלוק הנתונים וכל המידע באות אבד.

$$Av = \text{sum}(\text{signal}) / \text{length}(\text{signal})$$

$$Av = -0.018057$$

נגדיר פרמטר m המציין את גודל קבוצת המיצוע. אם m=8 אזי מספר קבוצות המיצוע יהיה n=128.

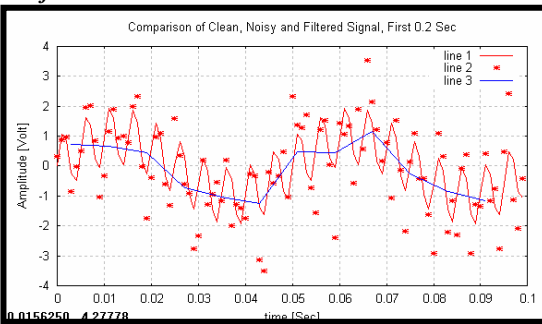
כלומר 1024 נקודות הדגימה באות המקורי יצטמצמו ל- 128 נקודות באות המסונן. בשפה של Octave זה נראה כך:

$$m=8;$$

$$\text{for } n=1:1024/m;$$

$$\text{sn}(n)=\text{sum}(\text{signal}((n-1)*m+1:m*n))/m;$$

$$\text{endfor}$$



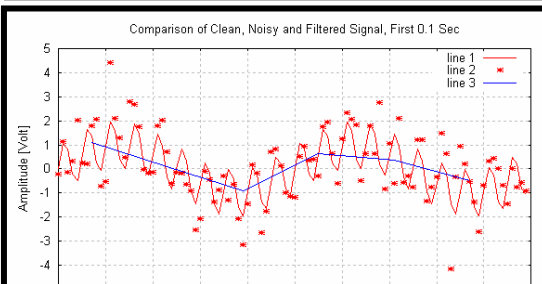
איור 3 : האות הנקי, אות + רעש והאות המסונן

האות המסונן מתואר גרפית באיור 3 יחד עם האות הנקי המקורי והאות הרגום הרועש (מסומן ב *). בחינה של הגרף מלמדת שהתדר הגבוה של האות יחד עם חלק מהרעש נעלמו. זהו אם כך פילטר מסוג Low Pass עם תכונות התלויות בעיקר בגודל קבוצת המיצוע.

להלן הפקודות לציור הגרף.

```
title("Comparison of Clean, Noisy and Filtered Signal, First 0.2 Sec");grid
```

```
plot(t(1:100),s(1:100),t(1:100),signal(1:100),'*','t',n(1:12),sn(1:12));
```



איור 4 : תוצאה של הגדלת קבוצת המיצוע

ניתן עכשיו בקלות לחזור על החישובים עם פרמטר m שונה. למשל 4 או 16. אינטואיטיבית – ככל ש- m גבוה יותר – הפילטר יותר 'חזק' כלומר תדר הקיטוע יהיה יותר נמוך.

הבה ונבחן את התוצאה על פי קריטריון Nyquist. בשנייה אחת היו לנו 128 דגימות כלומר פי 6.4 מהתדר הנמוך באות המבחן. ואכן התדר הגבוה נעלם לחלוטין ואילו התדר הנמוך (20Hz) שוחזר טוב. אם נבחר בקבוצת מיצוע יותר גדולה – למשל m=16, יהיו לנו 64 דגימות בשנייה ולפי הקריטריון נוכל לשחזר תדר של עד 32Hz איור 4 הוא למעשה חזרה על איור 3 עם קבוצת מיצוע בת 16 איברים.

כאן כבר ניתן להבחין בניחות של האות ב 20 Hz.

3.2. ממוצע צועד

שיטת המוצע הצועד היא למעשה שכלול של שיטת הממוצעים החלקיים. השיטה מבוססת על מילוי 'מחסנית' בגודל m . כאשר המחסנית מתמלאת, מחשבים את ממוצע התוכן שלה והוא מייצג את הערך של האות ברגע שהמחסנית התמלאה. עם קבל הערך הבא, גורעים מהמחסנית את הערך הדגום הישן ביותר ומחשבים ממוצע חדש. לדוגמה – אם נבחר גודל מחסנית 8, אזי 7 הדגימות הראשונות לא תוצאנה תפוקה בכלל. לכשתתקבל הדגימה השמינית, יפיק הפילטר את הממוצא הראשון שיוחס לזמן קבלת הדגימה השמינית. מהדגימה השמינית ואילך – יוציא הפילטר ערך מחושב עבור כל דגימה שתתקבל. כלומר – בניגוד לפילטר הקודם – רק $(m-1)$ דגימות ראשונות אוכדות בתהליך וכל היתר מפיקות ערך מחושב. גם הפילטר הזה הוא בעל אופי של Low Pass אבל אופי הפיגורים שלו מתאים יותר למטרות בקרה.

3.3. סימולציה מתימטית

פילטרים מסוג צ'ביצ'ב או בוטרורות על כל סוגיהם (ולמעשה כל הפילטרים המוכרים לנו כאוסף סלילים קבלים ונגדים) ניתנים לדימוי מתימטי כמנה של שני פולינומים B ו- A . מקדמי הפולינומים ודרגתם קובעים את סוג הפילטר (Low Pass, High Pass, Band Pass) וכן תכונות כמו גליות וניחות. Octave מאפשרת בניית פילטרים בצורה הבאה:

```
[b,a] = butter(n, Wc)
```

```
low pass filter with cutoff pi*Wc radians
```

```
[b,a] = butter(n, Wc, 'high')
```

```
high pass filter with cutoff pi*Wc radians
```

```
[b,a] = butter(n, [Wl, Wh]
```

```
band pass filter with edges pi*Wl and pi*Wh radians
```

```
[b,a] = butter(n, [Wl, Wh], 'stop')
```

```
band reject filter with edges pi*Wl and pi*Wh radians
```

(שים לב - π מסומן ע"י π ואילו ω מסומן ע"י W)

כאשר B ו- A הם וקטורים של מקדמי הפולינומים במונה ובמכנה.

הביטוי בסוגריים המרובעים $[Wl, Wh]$ הוא וקטור בעל 2 איברים. הראשון Wl מציין את תדר הקטעון התחתון והשני Wh מציין את תדר הקטעון העליון. שימו לב כי ללא ציון של תפקוד, הפילטר הוא או "Low Pass" או "Band Pass".

גבולות התדרים הללו קשורים לקצב הדגימה של האות המעובד. במקרה שלנו, קצב הדגימה הוא 1024 דגימות לשנייה ולכן התדר המירבי יהיה 512 הרץ.

ננסה להפיק את התדר הנמוך מתוך האות ע"י פילטר "Low Pass".

התדר המבוקש הוא $20/512$ מהתדר המירבי כלומר ~ 0.04 . נבחר פילטר מסדר 5 עם תדר קיטעון (יחסי) של 0.04 ע"י הפקודה:

```
[b,a]=butter(5,0.04;
```

ומקדמי הפולינומים המתקבלים הם:

```
B =1.0e-06*[ 0.80424, 4.02118, 8.04236, 8.04236, 4.02118, 0.80424]
```

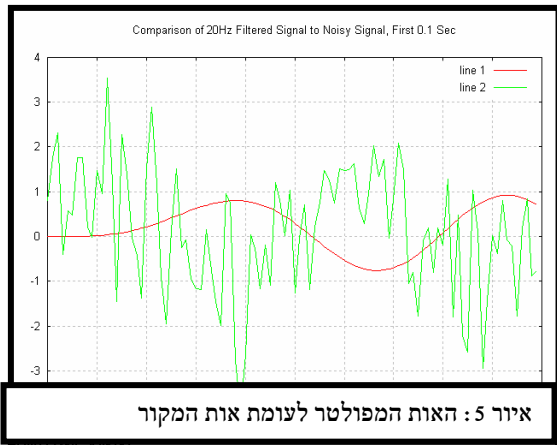
```
A =[1.00000, -4.59342, 8.45512, -7.79492 , 3.59890, -0.66565]
```

עיבוד האות נעשה ע"י הפקודה:

```
filtered=filter(B,A,signal);
```

האות מתקבל כאן בתוך משתנה בשם `filtered` ויש לנו ענין להשוות אותו לאות המקורי. שוב – הטקסט שיעשה לנו את זה מצורף להלן:

```
title("Comparison of 20Hz Filtered Signal to Noisy Signal, First 0.1 Sec");grid
plot(t(1:100),filtered(1:100),t(1:100),signal(1:100));
pause
```



הגרף עצמו (איור 5) מראה כמה תופעות הדורשות הסבר:

התופעה הראשונה הנראית כאן היא פיגור פאזה בסדר גודל של 180° . למרבית השימושים (למשל לשמע) אין לפיגור הזה משמעות מעשית. אבל אם היינו רוצים להשתמש באות למטרות בקרה, אזי פיגור הפאזה היה גורם שיש להתחשב בו. מתימטית – אין בעיה להריץ את הפילטר פעמיים – פעם קדימה ושוב פעם בסדר הפוך של הדגימות. פיגור הפאזה לעומת האות המקורי יעלם לחלוטין. כמובן שהתהליך כולו נמשך פי שניים יותר זמן, כלומר בעולם האמיתי נקבל את המידע המפולטר בפיגור זמן גדול פי שניים מאשר בהרצה בודדת של הפילטר.

התופעה השנייה הברורה באיור 5 היא נקודת התחלה לא זהה לאות האמיתי ולאות המפולטר. יש לזה הסבר

מתימטי ארוך ומסובך אבל ההסבר הפשוט הוא שפילטר מתימטי מוגדר לאות אינסופי בזמן. במקרה שלנו אנחנו מושפעים ממה שקרוי בשפת המתימטיקאים 'תנאי שפה'. ואומנם – ניתן להגדיר לפילטר נקודת התחלה או 'תנאי שפה' זהים לגודל האות המקורי כדי להמנע מחוסר ההתאמה בהתחלת הפילטר.

ישנן לפחות 2 שיטות נוספות לתקן את תנאי השפה:-

- התעלמות מתחילת האות המפולטר.
 - טיפול באות המקור הדגום ע"י 'חלון' – שם כולל למשפחה של מקדמים הכופלים את האות המקורי הדגום כך שיתחיל ב-0 ויגמר ב-0.
- קיימים מספר 'חלונות' סטנדרטיים המתאימים לעיבוד אות – נכיר אחד מהם בפרק הבא.

3.4 סינון במישור התדר

סינון במישור התדר הוא למעשה הרחבה של הפילטר הפולינומילי מהפרק הקודם. הפילטר הפולינומילי מתאפשר ע"י העתקת הדגימות של האות למישור מתימטי אחר. גם במקרה של סינון במישור התדר מתבצעת העתקה של האות הדגום ממישור הזמן למישור התדר.

הכלי לביצוע העתקה הזו נקרא 'המרת פורייה' על שם המתימטיקאי הצרפתי שהגה את הרעיון. בימיו של פורייה, העולם עדיין היה אנלוגי ורציף ולהמרת פורייה היה שימוש אקדמי מפואר לאותות תיאורטיים אינסופיים על הלוח בסורבון.

התפתחות הפיסיקה והמתימטיקה והשימוש הנרחב בתוצריהם הולידו את הצורך בהמרה שימושית ומהירה של אותות דגומים דיגיטליים. כך הגיעו לעולם 'המרת פורייה מהירה' (Fast Fourier Transform- FFT) וגם DFFT שזה אותו דבר אבל לאותות דגומים.

מבין כל הפילטרים שהוזכרו כאן – זה הפילטר החזק ביותר למטרות המוכרות ע"י חובבי הרדיו. חלק מהמקמ"שם המודרניים כולל תצוגה ספקטרלית של התדר הנקלט (למעשה זה הספקטרום של רמת ה-IF במקלט) ואשר לטענת המפתחים מסייע לבודד הפרעות, להעריך את עוצמת האותות ועוד.

בדוגמאות שאביא בפרק האחרון אראה שימוש פשוט בכרטיס הקול במחשב מגובה בתוכנת פילטר ספקטרי לסינון אות שמע בצורה המתאימה לחובב.