

# 高萩・日立アンテナサンプリングデータ解析におけるGPUを用いたFFT処理高速化

第15回水沢VLBI観測所ユーザーミーティング  
2017/11/3-4 @ 奥州市・北ホテル

秋田谷 洋(\*1,\*2)、米倉 覚則(\*1)、齋藤 悠(\*1)、百瀬 宗武(\*1)



(\*1) 茨城大学宇宙科学教育研究センター(秋田谷は2017/9まで所属)

(\*2) 埼玉大学大学院理工学研究科 戦略的研究部門X線・光赤外線宇宙物理領域

発表者連絡先

akitaya@mail.saitama-u.ac.jp



## 1 はじめに

- 茨城大学が運用する日立・高萩口径32mアンテナでは、観測時系列データのサンプリングに、K5/VSSP32サンプラーユニットを用いている。
- サンプリングデータのFFT処理・パワースペクトラム導出には、NICT提供の解析ソフトウェア群の一つspeana2.cを用いている。
- 従来のspeana2.cでは、FFT処理部にCPUで実行するC言語FFTサブルーチンを使用している。このFFT処理部分をGPU(Graphic Processing Unit)を呼び出すルーチンに変更し処理の高速化を図った。
- 試験用計算機で、1/16秒約  $4 \times 10^6$  点サンプリングデータ  $\times$  4800回(300秒積分に相当)のFFTスペクトル解析・平均化を行ったところ、解析所要時間は、CPU利用時の1089秒に対して、GPU使用時に最短274秒(比率約1/4)と大幅に短縮された。
- FFT計算部のGPU処理化は、スペクトル解析の時間短縮の上で極めて有効である。



図1: 高萩(左手前)・日立(右奥)口径32m アンテナ

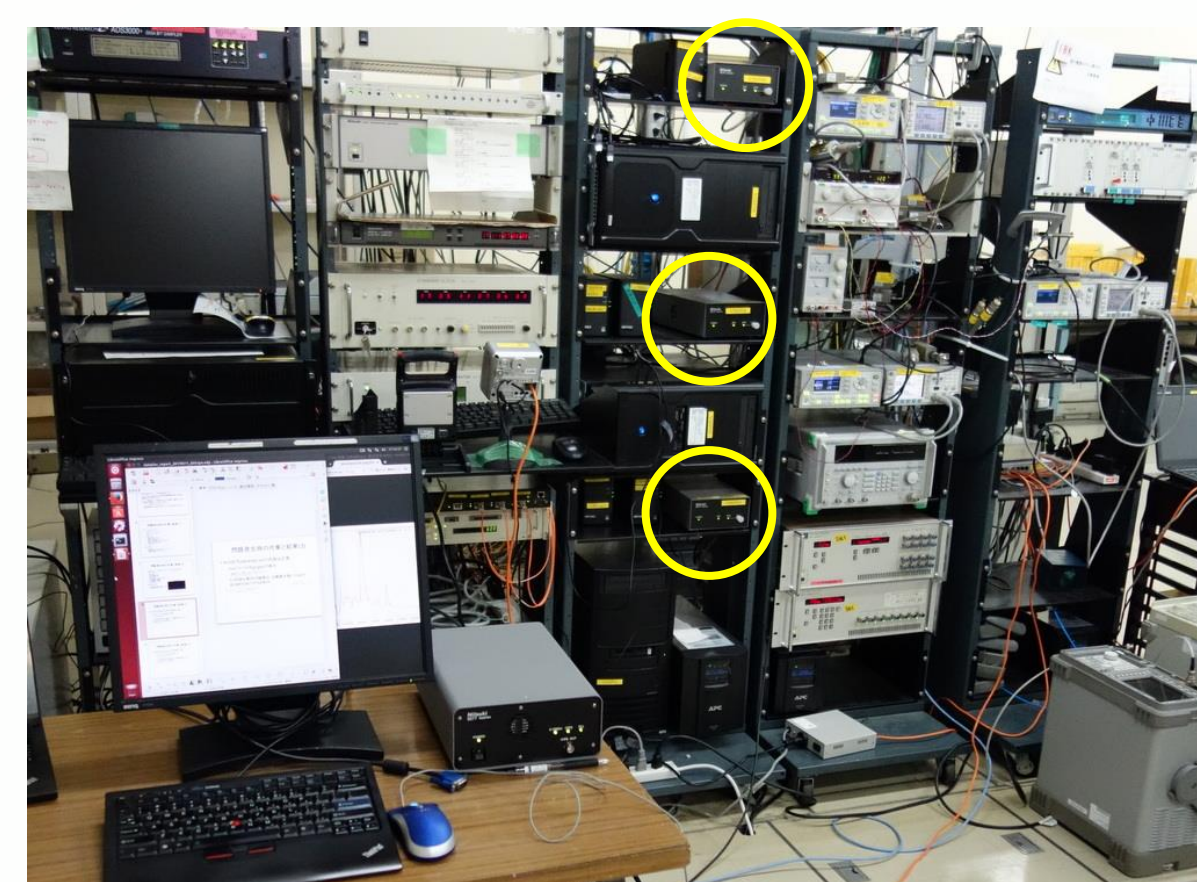


図2: サンプラー制御機器群とK5/VSSP32(丸印)

## 2 GPUと計算ライブラリについて

- GPU=Graphic Processing Unit: 主にグラフィック描画を主目的とした、数千コアの低機能計算機から成るユニット。(NVIDIA社解説参照; 図3)
- NVIDIA社製のPCIボードを中心に、安価で、かつ一般のPCにもよく搭載されている(購入者が意図せず載っている場合も多い)。
- NVIDIA社から、同社のGPUボード用にグラフィック処理、流体計算、FFT(=cuFFT)など、科学技術計算を含む多様なC言語用ライブラリ・APIが提供されている(=CUDA: Compute Unified Device Architecture: クーダ)。
- 既存のC言語ルーチンの置き換えが容易。解説文書も充実。
- 配列計算やFFT処理など、単純な並列計算に効力を発揮。並列計算用のコード書きの知識がなくても、通常のC言語コードにライブラリのAPIを埋め込めば、ライブラリで自動的に適切な処理に変換してくれる。
- 今回扱った茨城大データのFFTに限らず、より一般的な用途で大量データの多数計算処理を行う場合にも高速化が期待できる。

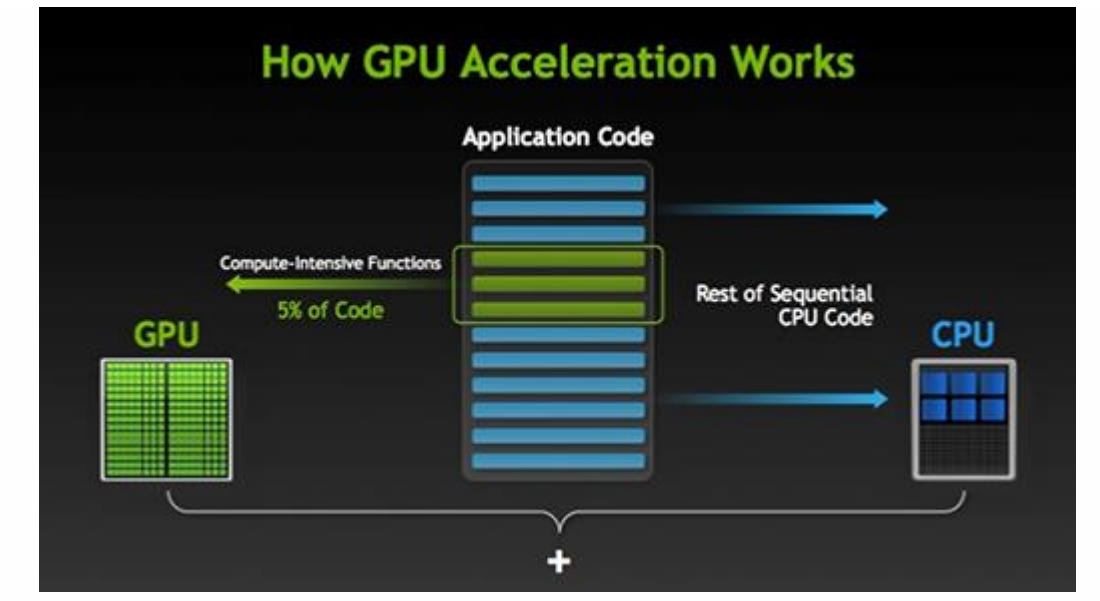


図3: CPUとGPUの計算処理分担のイメージ(NVIDIA社 web pageより)

## 3 FFT処理における課題と試験

### (1) サンプラー以降の観測データ取得の流れ

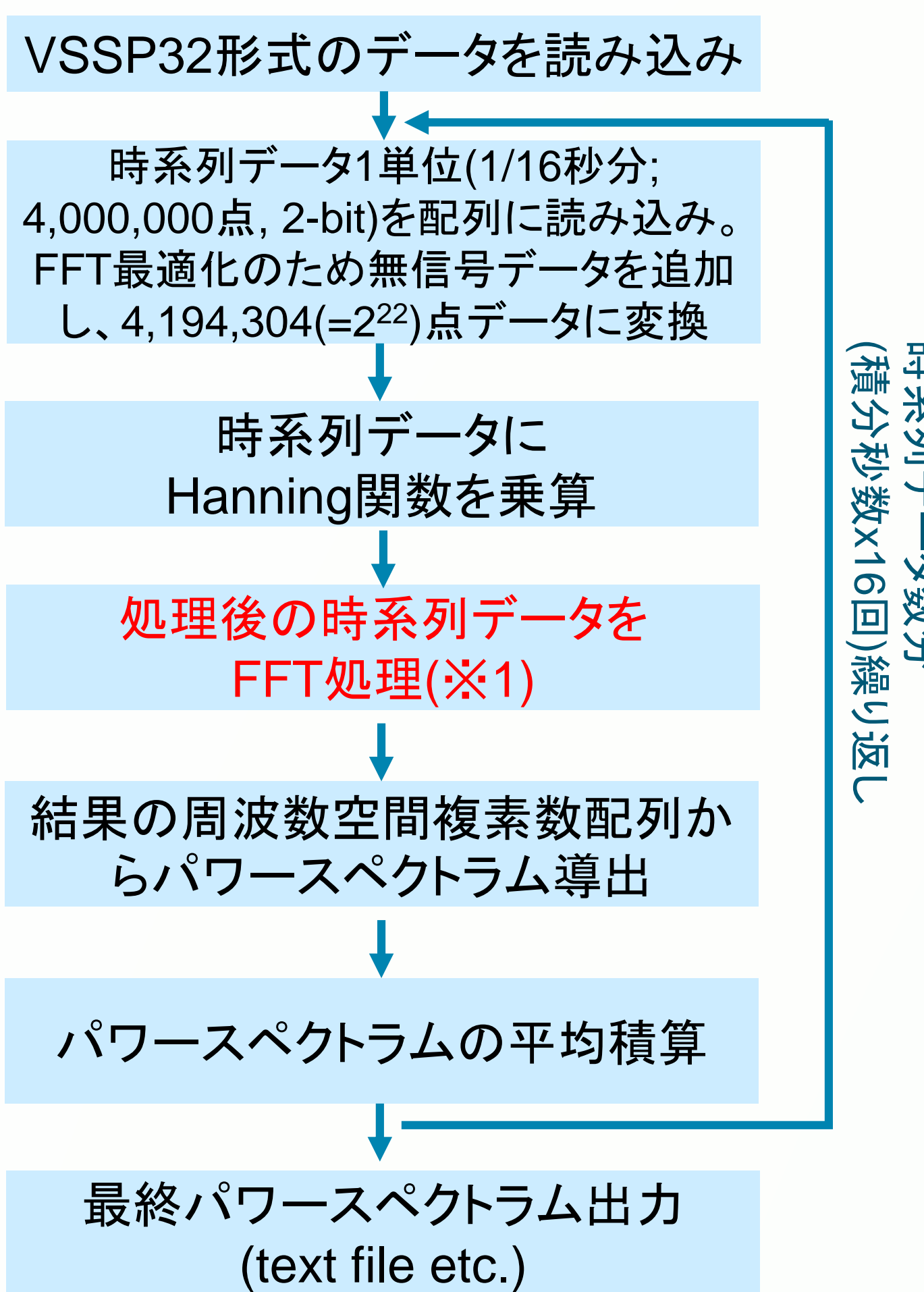
1. アンテナからダウンコンバートされた信号をVSSP32に入力。
2. 表1の条件で時系列データをサンプリング
3. 専用フォーマットのデータファイル(VSSP32形式)で保存

表1: データ取得のパラメーター

サンプリング速度	64 MHz
A/D 変換分解能	2-bit
1時系列当たりのサンプリング数	$4 \times 10^6$
1秒当たりの時系列データ数	16

### (2) データ解析(speana2.c内)の流れ

表2: speana2.cの解析の流れ



時系列データ数分(積分秒数 $\times$ 16回)繰り返し

### (4) 試験環境

- ・ 計算機は表3、図4解析に用いたデータは表4、図5の通り。

表3: 試験計算機環境

PC spec	core-i7 4790 (4 Cores/HT/3.6GHz) / 4GB memory
Graphic Board	NVIDIA GeForce GTX
OS/Distribution	Linux 3.13.0-117-generic / Ubuntu 14.04.5 LTS (x86_64)
CUDA Version	8.0
IP-VLBI ソフト	Ver. 2017-03-10
Speana2.c	Ver. 2017-02-22
FFTW3 library	Ver. 3.3.3-7

表4: 観測データ諸元

天体	W49N
波長帯	22235 MHz 中心、32 MHz 幅 (22219-22251 MHz) 水メーザー
サンプリングパラメーター	表1と同一
積分時間	300 sec

### (3) 現状の問題点と本研究の目的

- 解析中、FFT処理部(表2: ※1部)に処理時間を要する。さらにこれを多数回繰り返すため、最終パワースペクトラム出力までのデータ解析時間が膨大にかかっている。  
現状、茨城大学の観測・解析に用いているPCでは、観測時間  $\ll$  解析時間
- FFT処理部を高速化することが必要

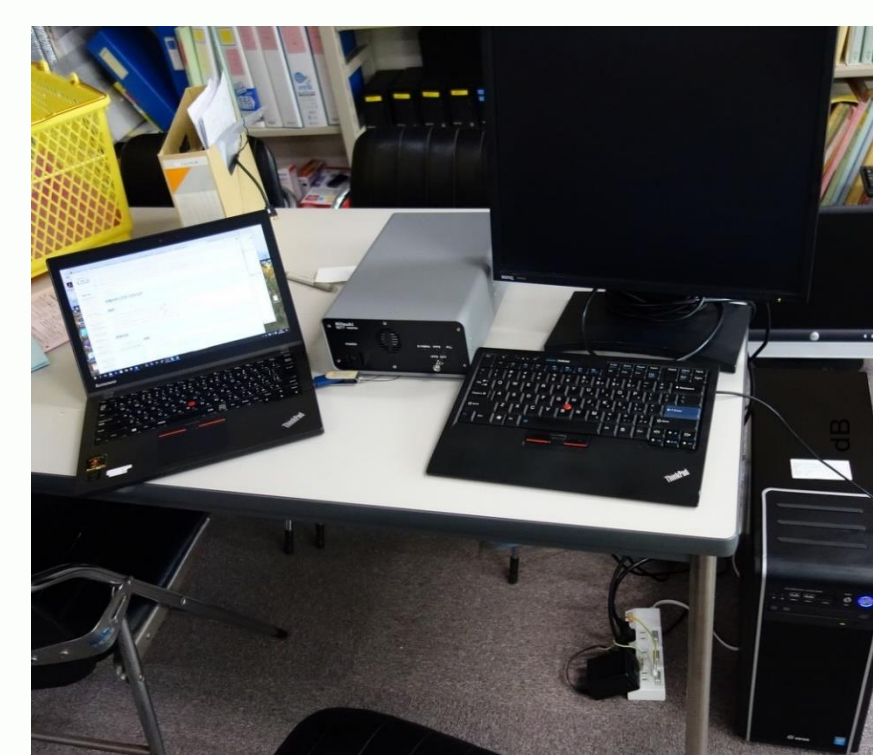


図4: 試験に用いたPC環境

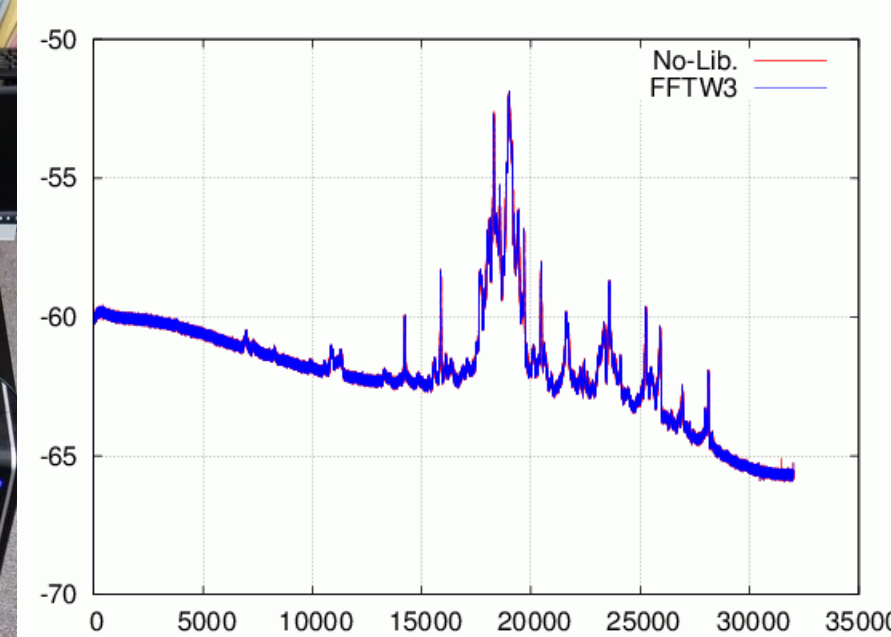


図5: 試験データの解析後スペクトラム(横軸はサンプリング率での周波数)

→ FFT処理部をGPUにて処理することで、パワースペクトラム解析速度を高速化する

## 4 結果

### (1) 解析所要時間比較

- 既存のspeana2.cについて以下の4パターンコードを用意し、試験データの処理速度の比較を行った(図6、表5)。

  1. FFT部に標準のC言語ルーチンを使用。(略号: nolib)
  2. FFTW3(CPU使用FFTを効率化したライブラリ)を使用。(同: fftw3); speana2.cのコンパイルオプションで選択可能
  3. speana2.cのコード全体はそのまま、FFT部のみをGPU用(CUDA cuFFT)に置き換え。(同: cuFFTW)
  4. speana2.cコード全体をGPU用に最適化してチューニング、FFTでGPU(cuFFT)使用。(同: cuFFT-pure)

- 処理のうち、主要な部分(データ読み込み、FFT処理、最終結果出力、それ以外)の処理速度もそれぞれ個別に測定した。

### (2) 結果

1. 全体の処理時間は、nolib、FFTW3、cuFFTW、cuFFT-pureの順に短縮された。最大で1089秒→274秒(比率約1/4)の大幅な短縮となった。
2. cuFFTWでは、nolib、FFTW3に対して、FFT処理部分の時間短縮が顕著であった。
3. cuFFTW→cuFFT-pureでは、FFT部の若干の処理時間短縮の他にFFT以外の処理部分(パワースペクトラム計算部)の処理時間短縮も見られた。
4. FFTが全体処理に占める割合は、GPU未使用時はFFT  $\gg$  全体であったが、GPU使用でFFT~全体となった。

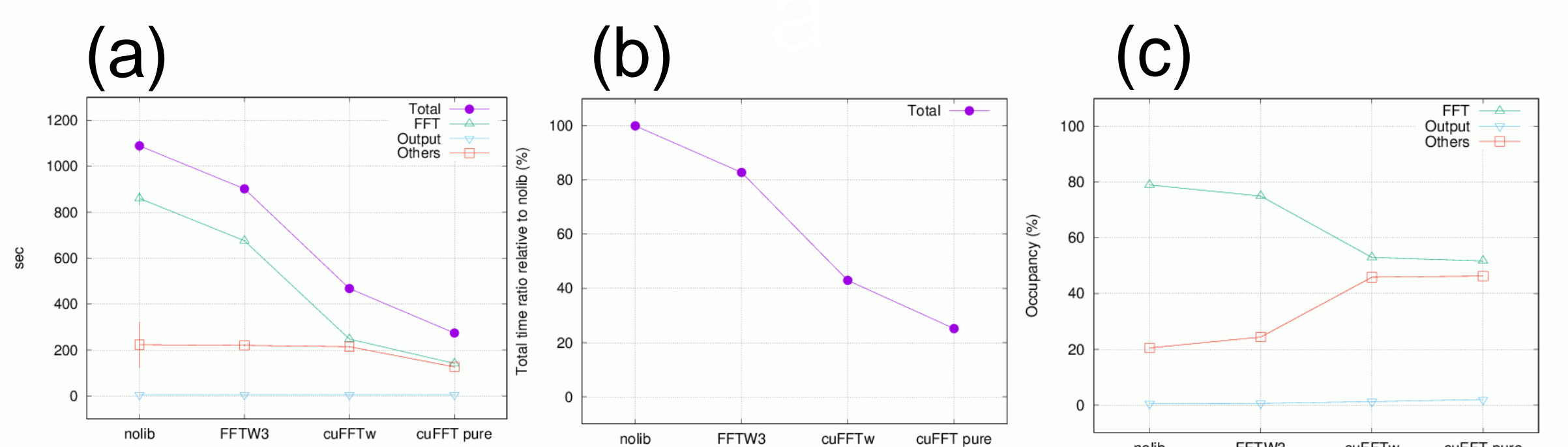


図6: (a)各ルーチンの実行所要時間、(b)nolibに対する時間比率、(c)FFT部とそれ以外の処理の時間比率

表5: 実行時間測定結果数値

FFT方式	図等での略称	全体 (s)	FFT (s)	出力 (s)	FFT・出力以外 (s)
K5/VSSP 付属FFTルーチン	nolib	1089 ± 28	860 ± 28	5 ± 0	223 ± 39
FFTW3	FFTW3	902 ± 3	676 ± 3	5 ± 0	220 ± 4
FFTW3コードのcuFFT変換	cuFFTW	488 ± 2	248 ± 0	5 ± 0	215 ± 2
cuFFT専用コード	cuFFT pure	274 ± 0	142 ± 0	6 ± 0	127 ± 1

### (3) 考察・まとめ

- 高萩・日立アンテナ観測データのサンプリングデータについて、PCを用いたFFT処理を行う場合、GPUを使用することで、大幅に高速化できる。
- GPU適用後のCPU処理はパワースペクトラム計算が大部分を占める。ここもGPU処理化することで、さらなる高速化が期待できる。
- 現在実際に観測で用いている古いPC(Core i7 4970x 3.5GHz/GeForce GT 610; 32-bit kernel)でも、処理時間はnolib=1430 sec→ cuFFT-pure=530 secと大幅短縮できた。  
→ 実運用すれば、現在、日立アンテナ一晩の標準的なデータ処理時間を28→10時間に短縮可能。(実運用には今後のソフトメンテの課題あり)