第7章 プロファイラを使ってボトルネックを抽出…初期化処理を26秒→219msに短縮！

CUDAプログラムの高速化にトライ

鈴木 量三朗

● 自作ライフ・ゲームをプロファイリングしてみる

エヌビディアが提供するNsight Systemsを使用すると，CUDAのプログラムのプロファイリングを取得できます．プロファイリングでは，プログラムがどのようなCUDA APIを呼び出して実行しているか，CUDAのGPU側のカーネルが動作しているかなど，プログラム上のボトルネックを解析するための情報を知ることができます．sudo権限で実行すると，OSで準備されている細かいプロファイリングも併せて取得できます．

作成して実行したファイルをnsysコマンドの引数にして実行することで，簡単にプロファイリングを取得できます．ここでは，第3部 第6章で作成したライフ・ゲームをプロファイリングしてみます．

ステップ①…プロファイリング実行

● コマンド実行とレポート出力

リスト1に実行時の様子を示します．ここではCUDAのプロファイルを取得するのが目的なので，sudoでの実行は必須ではありません．nsysコマンドを実行すると，プロファイリングのための準備をした後に指定されたcuda-lifeを実行します．プログラムの終了後，後処理をして指定されたプロファイル用のレポート・ファイルを出力します．

プログラムからの標準出力では，初期化処理が7msで終了し，26秒以上待たされた後に，実際のlifeが実行されます．実行が終了すると，カーネル側の出力であるprintfで指定した"life game start!!"が表示されました．

標準出力への情報出力は間違ってはいないものの，CPU側とGPU側で同期を取っているわけでもなく，実際にどのように実行されているかも分かりません．

● レポートの内容

プロファイルのレポートを見てみます．表示はnsys-uiコマンドで行います．

図1に示すのは，Timeline Viewというデフォルトの表示です．CUDA HWがGPU側，ThreadsがCPU側の動きをそれぞれ示しています．

リスト1　Nsight Systemsを使ってライフ・ゲームをプロファイリングした結果

```
$ nsys profile -t cuda,nvtx -o report7 ./cuda-life↵
…(中略)…
Collecting data...
処理開始
初期化終了： 7ms
処理開始： 26637ms
処理終了： 26637ms
life game start!!
最終シンク口終了： 26732ms
Generating '/tmp/nsys-report-41a0.qdstrm'
[1/1] [========================100%] report4.nsys-rep
Generated:
    PyTorch/cuda12.6/Work/StudyCUDA/Life3/
                                    report4.nsys-rep
```

▶ init_lifeの実行が99.7%を占める

CUDA HWをクリックしてツリーを開くと，さらに詳細が表示されます．［All Streams］も展開すると，Stream 13とDefault stream 7の2つのストリームが表示されました．Stream 13でinit_lifeを実行していて，99.7%の実行時間を占めていることが分かります．

▶ cudaStreamSynchronizeに時間がかかっている

次にCUDA APIを見てみます．右クリックでメニューを表示し，［Show In Events］を選択すると，下のペインのEvents ViewにAPIの呼び出した時刻と実行時間が表示されます．init_lifeが実行され，幾つかのAPIが実行された後，cudaStreamSynchronizeの実行に26秒以上かかっています．init_lifeを呼び出し，その処理が終わるまで待っているので時間がかかっていることが分かってきました．

ステップ②…初期化処理の高速化

● 乱数の初期化に時間がかかっている

どうやら乱数の初期化に時間がかかっているようです．