Using Intel Xeon Phi for Solving Non-Stationary Linear Programming Problems

Leonid Sokolinsky, Irina Sokolinskaya
South Ural State University (national research university)

The reported study has been partially supported by the RFBR according to research project No. 17-07-00352-a, by the Government of the Russian Federation according to Act 211 (contract No. 02.A03.21.0011) and by the Ministry of Education and Science of the Russian Federation (government order 1.9624.2017/7.8).
Non-stationary large-scale linear programming problem

\[
\max \left\{ \langle c_t, x \rangle \mid A_t x \leq b_t, x \geq 0 \right\}
\]

- \( x \in \mathbb{R}^n \)
- \( A_t \) – matrix \( m \times n \)
- \( b_t \) – \( m \)-dimensional vectors
- \( c_t \) – \( n \)-dimensional vector
- \( t \in \mathbb{R}_{\geq 0} \) – time

\( n, m > 10^6 \)

*Period of input data change* < \( 10^{-2} \) sec.
Asset-Liability Management

Dynamic linear programming problem
- 1.7 billion inequalities
- 5.1 billion variables

High Frequency Trading

- Dimension: $10^5$-$10^6$
- Number of inequalities: $10^6$-$10^7$
- Period of input data change: $10^{-2}$ - $10^{-3}$ sec.
NSLP Algorithm
(Non Stationary Linear Programming)

Algorithm phases:

• **Quest** – find point \( \bar{z} \in M_t \)

• **Targeting** – moving point \( \bar{z} \) in such a way that the solution \( \bar{x} \) of LP problem remains permanently in an \( \varepsilon \)-vicinity of \( \bar{z} \)

\[
A_t x \leq b_t \iff x \in M_t
\]
NSLP Algorithm
(Non Stationary Linear Programming)

Algorithm phases:

• **Quest** – find point $\bar{z} \in M_t$

• **Targeting** – moving point $\bar{z}$ in such a way that the solution $\bar{x}$ of LP problem remains permanently in an $\varepsilon$-vicinity of $\bar{z}$

$$A_t x \leq b_t \iff x \in M_t$$
**Quest Phase (Finding $\bar{z} \in M_t$)**

Conventional methods for solving a system of linear equalities can not give a solution to the problem $A_t x = b_t$ because of its non-stationarity.

\[ A_t x \leq b_t \iff x \in M_t \]

\[ A_{t'} x \leq b_{t'} \iff x \in M_{t'} \]
$M$ – convex bounded set

$\varphi \in \{R^n \rightarrow R^n\} – M$-fejerian map if

$\varphi(y) = y, \forall y \in M$;

$\|\varphi(x) - y\| < \|x - y\|, \forall y \in M, \forall x \not\in M.$
Fejerian Map for *Quest* Phase

\[ \varphi_t(x) = x - \frac{1}{m} \sum_{i=1}^{m} \max\{\langle a_{ti}, x \rangle - b_{ti}, 0\} \cdot a_{ti} \]

- \( a_{ti} \) – i-th line of matrix \( A_t \)
- \( b_{t1}, ..., b_{tm} \) – elements of column \( b_t \)
- \( m \) – number of lines in \( A_t \)
- \( t \) – time
Fejerian Process

\[ \varphi^s(x) = \varphi \ldots \varphi(x) \]

\[ x_0 \in \mathbb{R}^n \]

\[ \left\{ \varphi^s(x_0) \right\}_{s=0}^{+\infty} \]

\[ x_i = \varphi^i(x_0) \]
«Self-guidance» of Fejerian Process
«Self-guidance» of Fejerian Process
«Self-guidance» of Fejerian Process
«Self-guidance» of Fejerian Process
«Self-guidance» of Fejerian Process
«Self-guidance» of Fejerian Process
Scalable Synthetic LP Problem

\[
\begin{align*}
Q_{\text{max}}(x) &= 2x_0 + 2x_1 + \ldots + 2x_{n-2} + x_{n-1} \\
\left\{ \begin{array}{c}
x_0 \\
x_1 \\
\vdots \\
x_{n-1}
\end{array} \right. \\
x_0 + x_1 + \ldots + x_{n-1} &\leq 200(n - 1) + 100 \\
x_0 + x_1 + \ldots + x_{n-1} &\geq 100 \\
x_0 &\geq 0 \\
x_1 &\geq 0 \\
\vdots \\
x_{n-1} &\geq 0
\end{align*}
\]
## Benchmarked Processors

<table>
<thead>
<tr>
<th>Code Name</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>KNC</td>
<td>Intel Xeon Phi SE10X (61 cores, 1.1 GHz; 4 threads per core)</td>
</tr>
<tr>
<td>KNL</td>
<td>Intel Xeon Phi 7290 (72 cores, 1.5 GHz; 4 threads per core)</td>
</tr>
<tr>
<td>2×Broadwell</td>
<td>2×Intel Xeon E5 2697v3 32 cores, 2.6 GHz; 2 threads per core)</td>
</tr>
</tbody>
</table>
Speedup & Parallel Efficiency

**KNC**

- n = 12 000
- n = 9 600
- n = 4 800

**KNL**

- n = 12 000
- n = 9 600
- n = 4 800
Questions?