Institute of Applied Mathematical Research, Karelian Research Center of the RAS

Task Scheduling in a Desktop Grid for Virtual Drug Screening

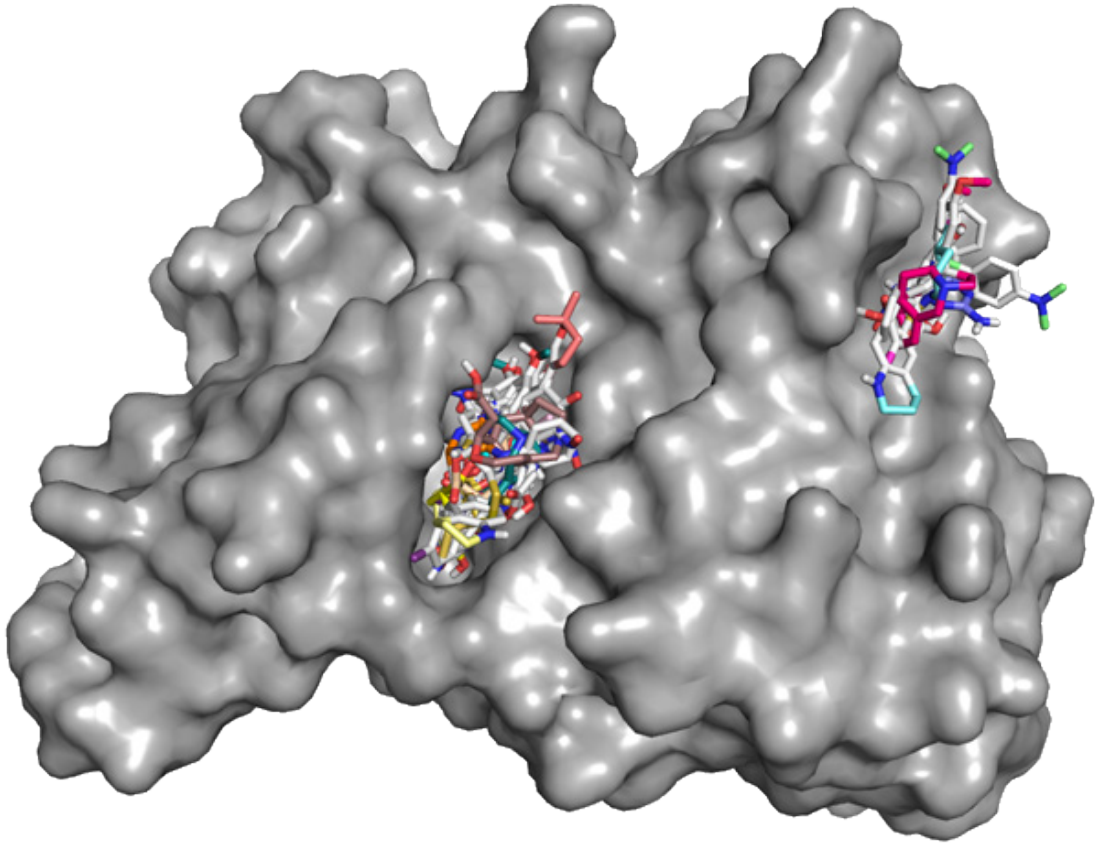
Natalia Nikitina, Evgeny Ivashko

Distributed Computing and Grid-technologies in Science and Education
Dubna, 5 July 2016

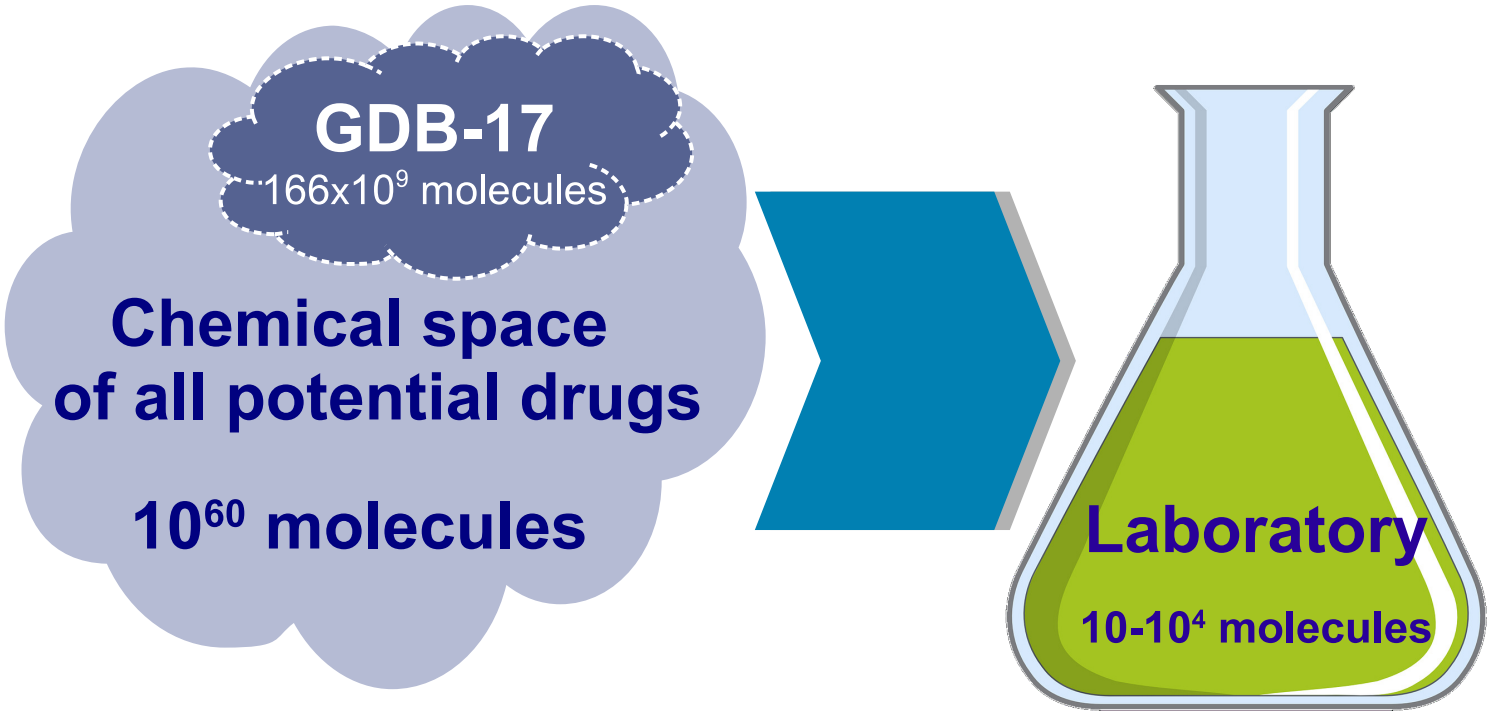
Virtual drug screening

Drug development

Drug development is a time-consuming and effort-consuming process which takes up to 7–15 years. The aim is to discover/synthesize a **ligand** molecule able to bind to a **target** molecule and influence the course of disease.

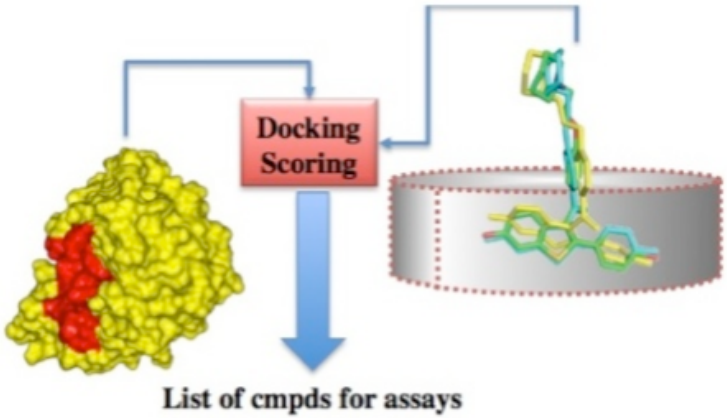


Virtual drug screening



To reduce input dataset for virtual screening down to manageable size, the chemical space is pre-filtered, leaving only representative compounds that promise to show desired biological activities.

Virtual drug screening



Virtual drug screening is the process of automated exhaustive search in the space of independent molecule models and selection of potential drugs among them. The aim is to reduce time and cost of the first stage of drug development process.

High-throughput computing is used to perform virtual drug screening.

Desktop Grid is a technology of utilizing idle CPU time of desktop computers over the Internet (volunteer computing) or over local area network of the organization (Enterprise Desktop Grid).

Task scheduling problem

- Huge size of libraries and computational cost of virtual drug screening;
- Pre-filtered libraries may be ineffective when studying new or rare diseases, as potentially good classes of molecules were filtered out. So the chemical diversity of results is limited.

(Using high-throughput computing) how to perform virtual screening and

- *... provide high diversity of results in limited time;*
- *... provide first successful results ASAP.*

Task scheduling problem

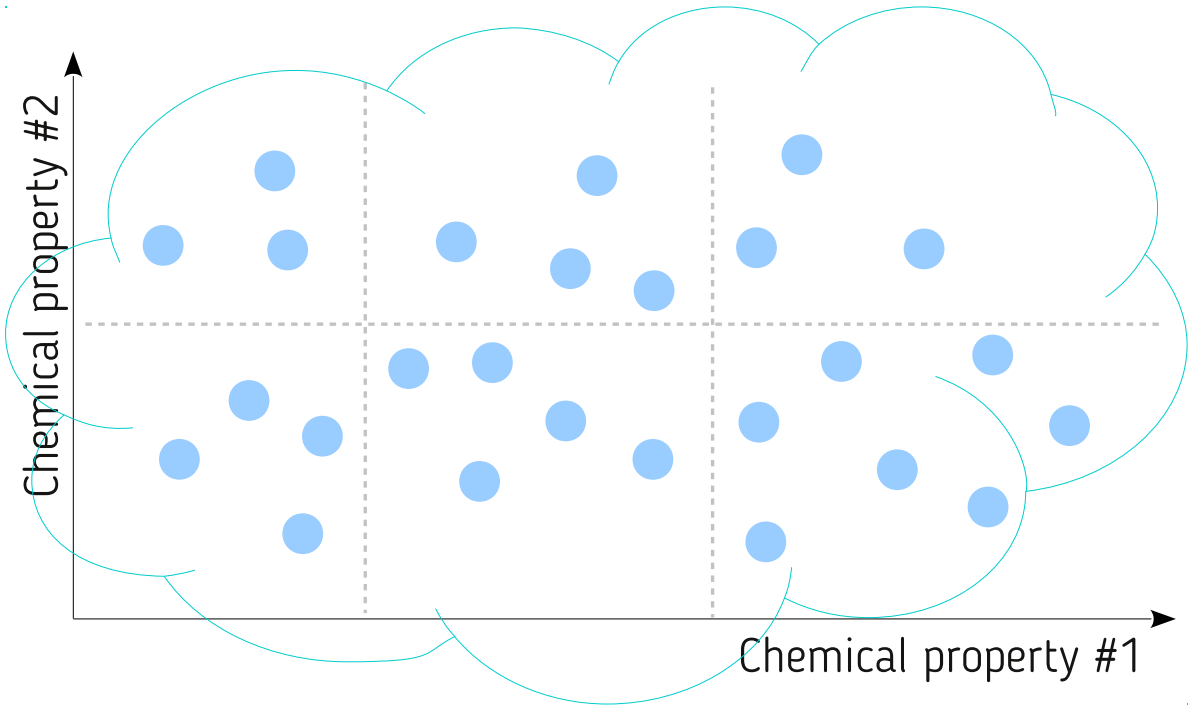


Figure 1: Library of molecules divided into blocks

Blocks priorities to search in:

- molecules of very simple/complex shape are less likely to become drugs;
- molecules highly similar to a known ligand are more likely to become drugs;
- etc.

Task scheduling problem

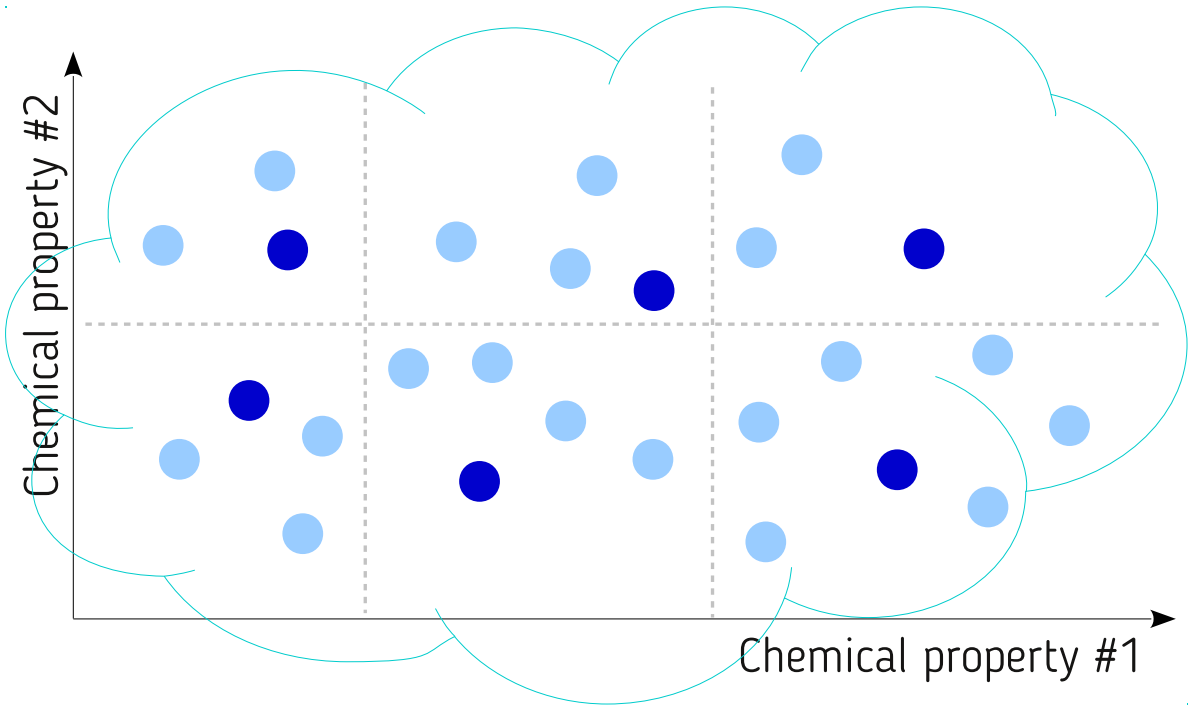


Figure 2: Library of molecules with selected results in each block

Task scheduling

- explore different blocks → obtain chemically diverse results in limited time;
- explore prospective blocks first → successful results in short time.

Mathematical model for task scheduling

$C_1, \ldots, C_M$ are the computational nodes (players), $M \geq 2$
T is the set of computational tasks

Mathematical model for task scheduling

$C_1, \ldots, C_M$ are the computational nodes (players), $M \geq 2$
T is the set of computational tasks

Mathematical model for task scheduling

$C_1, \ldots, C_M$ are the computational nodes (players), $M \geq 2$
T is the set of computational tasks

p_j is the expected fraction of useful results in block T_j
$\sigma_j = \frac{p_j}{p_1+\ldots+p_M}, 0 \leq \sigma_j \leq 1$ is the priority of block T_j

Mathematical model for task scheduling

ops_i is the performance of the computational node (number of operations per second)

θ_j is the complexity of a computational task (number of operations)

τ is the considered time interval

n_j is the number of players that have chosen block T_j

$\delta(n_j) = \frac{M+1-n_j}{M}$ is the congestion coefficient of block T_j

$U_{ij} = \sigma_j \delta(n_j) \frac{ops_i}{\theta_j} \tau$ is the utility of node C_i which chooses block T_j

$\vec{s} = (s_1, \ldots, s_M)$ is the strategy profile (blocks selected by each player)

Mathematical model for task scheduling

$$\Gamma = \langle C, T, U \rangle$$

- The game Γ always possesses Nash equilibria in pure strategies. *(R. Rosenthal. "A Class of Games Possessing Pure-strategy Nash Equilibria". International Journal of Game Theory, Vol. 2, I. 1, pp. 65–67, 1973)*
- Best- and better-response dynamics converge to Nash equilibrium in polynomial time. *(S. Ieong et al. "Fast and Compact: A Simple Class Of Congestion Games". AIII Proceedings, pp. 1–6, 2005)*

Mathematical model for task scheduling

$\vec{s} = (s_1, \ldots, s_M)$ is a strategy profile (blocks selected by each player)

Each player maximizes his **personal utility** $U_{ij}(\vec{s})$

$SOC(\vec{s})$ is the **social utility**. In the worst equilibrium, SU is minimal over all equilibria. In the best outcome, SU is maximal.

$OPT(\Gamma) = \max_{\vec{s}} SOC(\vec{s})$ is the maximal social utility in game Γ

PoA = Worst equilibrium SU / Best outcome SU

PoA (the Price Of Anarchy) expresses the efficiency of independent players' choice without a centralized decision maker.

$PoA(\Gamma) = \min_{\vec{s} \text{ is a NE}} \frac{SOC(\vec{s})}{OPT(\Gamma)}$ is the price of anarchy in game Γ

Mathematical model: social utility

Social utility for the computational system

Amount of useful computations (sum of utilities) *(N. Nikitina, E. Ivashko. "The Price of Anarchy in a Congestion Game for Drug Discovery" [in Russian]. Extended abstracts of the IX International Petrozavodsk Conference, Probabilistic Methods in Discrete Mathematics, May 30 – June 3, 2016, Petrozavodsk, Russia, pp. 67–69)*:

$$SOC(\vec{s}) = \frac{ops \times \tau}{\theta \times M} \sum_{i=1}^{N} \sigma_i n_i (M + 1 - n_i)$$

Task scheduling game was analytically investigated for the case of two blocks, $N = 2$, and M identical players. If $\sigma_1 \geq \sigma_2$, then $PoA > PoA_e$,

$$PoA_e = \begin{cases} \frac{8}{3(M+2)} \text{ if } M \text{ is even and } \sigma_1 > \frac{M}{M+1}, \\ \frac{9}{4\left(M+1-\frac{1}{M+1}\right)} \text{ if } M \text{ is odd and } \sigma_1 > \frac{M}{M+1}, \\ \frac{4\lfloor \sigma_1(M+1) \rfloor (M - \lfloor \sigma_1(M+1) \rfloor)}{M(M+2)} \text{ if } M \text{ is even and } \sigma_1 \leq \frac{M}{M+1}, \\ \frac{4\lfloor \sigma_1(M+1) \rfloor (M+1 - \lfloor \sigma_1(M+1) \rfloor)}{(M+1)^2} \text{ if } M \text{ is odd and } \sigma_1 \leq \frac{M}{M+1}. \end{cases}$$

Mathematical model: social utility

Social utility for the laboratory

Expected quality of virtual screening results:

$$SOC(\vec{n}) = \frac{ops \times \tau}{\theta \times M} \left(\chi(\vec{n}) \sum_{i=1}^{N} \sigma_i n_i - \frac{\max(\sigma_i n_i)}{2} + \frac{\min(\sigma_i n_i)}{2} \right)$$

$\vec{n} = (n_1, \ldots, n_N)$ is a load vector;
$\chi(\vec{n})$ is the number of blocks chosen by at least one node.

If $N = 2$ and $\sigma_1 \geq \sigma_2$, then the PoA is estimated as $PoA \geq PoA_e$, where

$$PoA_e = \begin{cases} \frac{55}{64} \text{ if } \sigma_1 < \frac{5}{8}, \\ \frac{10}{5+2\sqrt{10}} \text{ if } \frac{5}{8} \leq \sigma_1 < \frac{M}{M+1}, \\ \frac{1}{3} \text{ if } \sigma_1 \geq \frac{M}{M+1}. \end{cases}$$

Comparison to other methods

Alternative methods to solve the problem

- Pre-filtering of the chemical space (clustering, Monte Carlo method, simulated annealing...)
  - Requires much knowledge about disease target and known ligands;
  - Omits potentially interesting compounds for rare or novel diseases;
  - Requires complex post-filtering of virtual screening results.
- Genetic algorithms with stochastic search *(C. Rupakheti et al. "Strategy To Discover Diverse Optimal Molecules in the Small Molecule Universe". Journal of Chemical Information and Modeling, 2015, 55(3), pp. 529–537)*
  - Requires redundant computations;
  - In general case, does not guarantee useful results in appropriate time.

Conclusion

Current results

We have proposed a mathematical model and a task scheduling algorithm for Desktop Grid that allow to ensure:

- Chemical diversity of virtual screening results;
- Early delivery of first results;
- Lesser overhead costs in algorithm implementation.

Thank you for your attention!