Математическая Теория Игр и её Приложения, т.10, в.3, с. 76–90

УДК 519.83
ББК 22.176

ТЕОРЕТИКО-ИГРОВАЯ МОДЕЛЬ СЕТИ ДОБРОВОЛЬНЫХ ВЫЧИСЛЕНИЙ

Илья А. Чернов*
Институт прикладных математических исследований
Карельского научного центра РАН
185910, Петрозаводск, ул. Пушкинская, 11
e-mail: chernov@krc.karelia.ru

В статье предложена простая игровая модель сети добровольных вычислений, в которой репликация заданий призвана снизить ущерб от злонамеренного искажения ответов. Атака злоумышленника посредством внедрения многочисленных узлов в сеть приносит некоторый доход от нарушения работы, тогда как сервер терпит убытки, приняв неверный ответ. Узлам приписана репутация, монотонно зависящая от числа правильных либо неразоблаченных ложных ответов. Получены оптимальные смешанные стратегии и показано, что средний выигрыш игроков зависит только от убытков сервера, репутации узлов и размера подсети узлов с данной репутацией. Получены оценки на затраты сервера на один ответ. Численные примеры показывают, что средние расходы сервера при использовании узлов с хорошей репутацией меньше, чем в случае, когда число внедренных узлов точно известно.

Ключевые слова: добровольные вычисления, desktop grid, антисаботаж, репутация.

---

* Поддержано грантами РФФИ 18-07-00628 и 16-07-00622

1. Введение

Проекты добровольных вычислений занимают важную нишу в области научных вычислений. Достаточно упомянуть такие популярные проекты, как SETI@home, положивший начало парадигме добровольных вычислений, Folding@home, Collatz Conjecture, Climate Prediction — полное количество насчитывает несколько десятков, а охват научных областей весьма широк. Разработано несколько программных систем для организации таких вычислительных сетей, наиболее популярной из них является BOINC[1] — на сайте проекта опубликован актуальный список активных проектов на базе системы.

Вычислительные сети из компьютеров общего назначения, соединенные коммуникационными сетями, называют грид-системами (Desktop Grid). Задачи, которые могут эффективно решаться, обычно состоят из большого числа сравнительно простых независимых между собой вычислительных заданий. В большинстве случаев это задачи перебора или поиска. В частности, определенное внимание уделяется изучению белковых молекул в интересах биологии и медицины. Помимо публичных проектов GPUGRID, DrugDiscovery@home, Folding@home, научные группы развертывают локальные вычислительные проекты, задействуя ресурсы одной или нескольких организаций. Примером может служить проект по виртуальному скринингу, описанный в [1].

Одной из проблем добровольных вычислений является обеспечение безопасности — в данном контексте, надежности полученных ответов. Добровольцы присоединяются к проекту анонимно или, во всяком случае, без серьезной аутентификации. В этом случае неизбежны злоумышленные враждебные действия, обусловленные либо желанием нажиться за счет организатора проекта, либо психологией человека в условиях безнаказанности («*Aditum nocendi perfido praestat fides*»). Вопросам борьбы со злоумышленниками уделяется значительное внимание, см., например, [2]. Одним из наиболее эффективных и часто используемых средств противодействия злому умыслу является репликация: расчет одного и того же задания на нескольких узлах со сравнением результатов. Вопросы распределения заданий среди узлов с учетом репликации изучены пока недоста-

[1] https://boinc.berkeley.edu/

точно; обзор проблем и результатов в этой области дан, например, в [3]. В ряде случаев достаточно дубликации заданий, чтобы защититься от злоумышленного искажения ответа добровольцем-одиночкой. Однако при массированной атаке требуются более сложные методы противодействия, в том числе учет репутации узлов, основанной на истории работы.

Теоретико-игровой подход к проблеме распределения заданий выглядит многообещающе, однако до сих пор изучен недостаточно. Хотя такие методы эффективны в условиях гетерогенной сети с постоянно меняющейся структурой, они необходимы для анализа ситуации противостояния организатора вычислений и атакующего злоумышленника, поскольку интересы этих лиц если и не противоположны, то существенно не совпадают.

В работе [4] предлагается игровая модель для вычисления множества независимых подзаданий, эквивалентных по значимости. Игроки — сервер и злоумышленник, который внедряет некоторое число зловредных узлов. Сервер стремится минимизировать долю ошибок, а злоумышленник — максимизировать. Приводятся два алгоритма для выбора оптимальных значений параметров сервером и доказана сходимость к равновесию по Нэшу, которое обеспечивает близкую к минимальной долю ошибок.

В статье [5] предлагается многошаговая игровая модель, в которой клиенты-игроки на каждом шаге выбирают — солгать или не солгать. При отклонении игрока от общей стратегии, игроки наказывают его на следующем шаге, даже себе в ущерб. Для случаев чистых и смешанных стратегий приводятся условия, при которых сервер получит корректный ответ с высокой вероятностью, и расходы сервера в этом случае. Работа [6] тех же авторов предлагает игровую модель вычислительного процесса в Desktop Grid, где игроки могут лгать — подделывать результаты. При этом они могут объединяться в коалиции, где каждый делает один и тот же выбор — лгать или не лгать. Цели сервера: максимизировать вероятность получения корректного результата и максимизировать выигрыш или минимизировать свои расходы. Сервер назначает реплики n клиентам и с некоторой вероятностью проверяет ответы (проверка затратна) либо доверяет им. Если проверяет, то награждает честных и штрафует нечестных, а если

не проверяет, то просто принимает ответ большинства и награждает их. Найдены аналитические выражения условий на значения параметров, при которых клиентам будет наиболее выгодно всегда быть честными, и показано существование единственного равновесия в чистых стратегиях, при котором сервер получает корректный результат в случае, если лгущих не окажется большинство. В работе [7] (того же коллектива) игровой подход применяется для обеспечения надежности добровольных вычислений в грид на основе BOINC. Узлы относятся к одному из трех типов: честные, возвращающие всегда верный ответ, злоумышленники, которые всегда лгут, и корыстные, преследующие свою выгоду. Сервер имеет возможности поощрять и наказывать узлы и использует функцию репутации для выбора стратегии. Рассматривается три функции репутации: стандартная BOINC, из литературы и авторская. Цель сервера — выявить злоумышленников и склонить корыстных к сотрудничеству, обеспечив тем самым достоверность ответа. Схема моделируется марковской цепью и показывается ее работоспособность при определенных условиях.

Следует различать две ситуации: активные или интеллектуальные узлы могут анализировать полученные задания, сравнивать их между собой, принимать совместные решения; либо же узлы не координируют свои действия, принимая решения независимо.

В [8] игровая схема используется для снижения нагрузки на сервер, а статья [9] посвящена игровой оптимизации задачи поиска.

Мы предлагаем простую модель вычислительной сети и игровую задачу на ее основе: организатор вычислений выбирает уровень репликации, жертвуя производительностью ради безопасности или наоборот, а злоумышленник — число лгущих узлов из числа внедренных. Разоблачение обмана приводит к потере репутации и, следовательно, к затратам на ее восстановление (это работа злоумышленника в пользу проекта). Однако успех в обмане приносит злоумышленнику некоторый доход. Предполагаем, что узлы злоумышленника действуют независимо.

2. Модель

Рассмотрим грид-систему (Desktop Grid) добровольных вычислений, состоящую из сервера и большого количества компьютеров (узлов), соединенных коммуникационной сетью. Узлы запрашивают у

сервера задания, выполняют и возвращают ответ на сервер. Решается задача поиска: каждое задание состоит в вычислении некоторого функционала на элементе множества и требуется найти элементы с малым (в определенном смысле) значением этого функционала.

Узел может вернуть неверный ответ, из-за ошибки или злонамеренно. Примером ошибки может служить сходимость алгоритма типа спуска к локальному минимуму или седловой точке. Мы рассматриваем только такие ошибки, которые могут быть выявлены повторным расчетом с другими начальными условиями. В целях безопасности сервер посылает одинаковые задания различным узлам. Это позволяет снизить риск принятия неверного ответа, посланного злоумышленником, а также выявить добросовестную ошибку. Разоблаченный узел может быть внесен в «черный список» и исключен из проекта. Однако, в этом случае будут исключаться также и допустившие добросовестную ошибку узлы. Желательно дать возможность реабилитации. Поэтому введем величину, называемую «репутацией», которая зависит от истории работы всех узлов; определим ее позже.

С точки зрения сервера ситуация выглядит следующим образом. Имеется конечное множество узлов, на которых задана функция репутации. Можно выделить срез из тех, на которых репутация не меньше некоторого значения и использовать в первую очередь их. Затем выделить новое множество узлов с приблизительно одним уровнем репутации, и так далее, пока все узлы не будут задействованы. Поэтому можно ограничиться подмножеством узлов, на которых репутация не ниже некоторого порогового значения (но и не слишком высока, так что можно считать ее постоянной). Решение, которое принимает сервер — это число реплик задания. Большее число реплик снижает вероятность ошибок, как добросовестных, так и злонамеренных, но пропорционально снижает производительность совокупности вычислительных узлов. Выявление ошибки снижает репутацию тех узлов, которые ее допустили.

Перед злоумышленником стоит задача внедрения своих вычислительных узлов в сеть и нарабатывание репутации. Злоумышленник получает выигрыш в случае, если удалось обмануть сервер, заставив его принять неверный ответ, однако несет затраты на внедрение в сеть и выполнение полезной работы.

Возникает игровая ситуация. При данном уровне репутации, сервер может либо ценой затрат проверять задания тщательно (высокий уровень репликации), либо дублировать задания или не проверять вовсе, ценой высокого риска. Злоумышленник может внедрить много узлов, а может внедрить несколько или ни одного, а также может сотрудничать с проектом, а может лгать. Очевидно, что знание стратегии противника дает решающее преимущество, и поэтому игра в чистых стратегиях не решается.

Рассмотрим две задачи. В первой ответы не различаются по ценности. Ложь злоумышленника заключается в искажении ответа. Ответы не могут совпасть случайно, а только лишь в результате систематической ошибки либо координированных злонамеренных действий. Во второй задаче ответы делятся на два класса: интересных, с низким значением функционала — они встречаются редко и известна оценка вероятности такого ответа; и неинтересных, которых большинство. Злоумышленник может лгать двояко: выдавая интересный ответ за неинтересный (в тех редких случаях, когда он получен) и наоборот. В обеих задачах выигрыши игроков рассматриваются в среднем.

За единицу стоимости примем среднюю стоимость расчета одного задания. Все узлы считаем идентичными. Несмотря на высокую степень гетерогенности Desktop Grid, такое предположение допустимо, поскольку потери производительности из-за дублирования работы существенно превосходят неоднородность производительности вычислительных узлов.

3. Постановка задач

Стратегия сервера — выбор уровня репликации при данном уровне r репутации узлов. Предполагается, что узлы с более высокой репутацией уже загружены работой. Полное число таких узлов обозначим $N + M$ — здесь N — честные вычислители, а M принадлежат злоумышленнику. Из $N + M$ доступных узлов ν получают одно и то же задание. Если все ответы совпадают, этот ответ принимается. Он может быть неверным, если все узлы принадлежат злоумышленнику. В этом случае к расходам добавляется штраф F, связанный, например, с упущенной прибылью, ударом по репутации, и т.п. Таким

образом, расходы на расчет задания составляют ν единиц, независимо от наличия вредительской деятельности и ее разоблачения. Если ответы различаются, то есть выявлена вредительская деятельность, все задания решаются на доверенном компьютере и, таким образом, все солгавшие узлы гарантированно разоблачаются. Затраты на этот контрольный расчет не учитываем. Вероятность получения неверного ответа в одном расчете обозначим p — очевидно, она зависит от числа внедренных злоумышленником узлов, а также от числа этих узлов, которые примут решение лгать.

Злоумышленник же принимает последовательно два решения. Во-первых, он внедряет некоторое количество M узлов в систему и, выполняя задания, повышает их репутацию до данного (рассматриваемого) уровня. На это он тратит $Is(r)$ единиц стоимости на каждый узел; здесь I — затраты на внедрение узла, а $s(r)$ — затраты на наработку репутации (пока об этой функции предполагаем лишь ее монотонность, положительность и $s(\infty) = \infty$). Величину I можно трактовать как коэффициент преобразования единицы стоимости сервера (число решенных заданий) в единицу стоимости злоумышленника — цена расчета одного задания. Во-вторых, располагая M внедренными узлами, он решает, сколько из них солгут. Мы рассматриваем простейшую ситуацию, при которой узлы не сравнивают полученные задания и не координируют действия. Пусть m узлов сообщают неправильный результат. Если их разоблачат, злоумышленник вынужден восстановить их репутацию, неся затраты в виде полезной работы на сервер. Если сервер поверил и принял ответ, то злоумышленник получает некоторую прибыль G.

Возможно три случая:

1. Все реплики попали на узлы злоумышленника (его деятельность не разоблачена) и неверный ответ принят сервером; вероятность этого события C_m^ν / C_{N+M}^ν, дополнительный расход сервера равен F, доход злоумышленника равен G.

2. Все реплики попали на честные узлы и принят правильный ответ; вероятность этого события C_N^ν / C_{N+M}^ν, дополнительный расход сервера и доход злоумышленника равны нулю.

3. Все остальные распределения заданий по узлам, когда в расчете

участвуют и честные, и нечестные узлы — последние разоблачаются, дополнительный доход сервера $ms(r)$ равен полезной работе, которая потребуется для восстановления репутации r для m внедренных и разоблаченных узлов, а злоумышленник совершает эту работу, затрачивая $Ims(r)$ единиц стоимости.

Тогда средние расходы сервера составляют

$$C = \nu + \frac{C_m^\nu}{C_{N+M}^\nu} F - \left(1 - \frac{C_m^\nu}{C_{N+M}^\nu} - \frac{C_N^\nu}{C_{N+M}^\nu}\right) ms(r), \tag{3.1}$$

а выигрыш злоумышленника равен, в среднем,

$$V = \frac{C_m^\nu}{C_{N+M}^\nu} G - \left(1 - \frac{C_m^\nu}{C_{N+M}^\nu} - \frac{C_N^\nu}{C_{N+M}^\nu}\right) Ims(r). \tag{3.2}$$

Используем соглашение: $C_n^m = 0$ при $m > n$. Имеет место оценка

$$V \le \frac{C_m^\nu}{C_{N+M}^\nu} G,$$

которая достигается при $\nu = 1$, $m = M$. В самом деле, V монотонно убывает по ν, а оценка по m получается следующим образом:

$$V \le \frac{MG}{N+M} + \left(\frac{M}{N+M} + \frac{N}{N+M} - 1\right) Ims(r) = \frac{MG}{N+M}.$$

Вместе с тем, возможен случай и $V < 0$; поэтому у злоумышленника нет доминирующей стратегии, поскольку стратегия невмешательства $m = 0$ гарантирует $V = 0$.

4. Решение игровой ситуации

Рассмотрим крайние случаи: $\nu = 1$, $\nu = N + M$ (этот случай, отказ от параллельности, эквивалентен проверке каждого расчета на доверенном вычислительном устройстве), $m = 0$ (отказ от враждебной акции) и $m = M$ (атака всеми силами). Получаем биматричную игру с матрицами выигрышей, указанными в таблицах 1 и 2.

Таблица 1. Матрица выигрышей сервера.

	$m = 0$	$m = M$
$\nu = 1$	-1	$-1 - \frac{MF}{N+M}$
$\nu = N + M$	$-(N + M)$	$Ms(r) - (N + M)$

Из таблицы ясно, что стратегия $\nu = 1$ у сервера доминирующая, если репутация r удовлетворяет неравенству

$$s(r) \le 1 + \frac{N-1}{M} - \frac{F}{N+M}. \tag{4.1}$$

Для узлов с такой репутацией серверу нет смысла проверять задания и игра решается в чистых стратегиях: $\nu = 1$, $m = M$. Иными словами, сервер отказывается от борьбы, осознавая, что результаты максимально подделываются.

Таблица 2. Матрица выигрышей злоумышленника.

	$m = 0$	$m = M$
$\nu = 1$	0	$\frac{MG}{N+M}$
$\nu = N + M$	0	$-MIs(r)$

Рассмотрим случай игры в смешанных стратегиях. Пусть сервер выбирает стратегию $\nu = 1$ с вероятностью p; тогда условие равновесия — это

$$p\frac{MG}{N+M} - (1-p)MIs(r) = 0,$$

то есть

$$p = \frac{Is(r)}{\frac{G}{N+M} + Is(r)} = \frac{Is(r)(N+M)}{G + Is(r)(N+M)}.$$

Эта вероятность определяется соотношение наживы G от обмана с одним ответом и затратами на внедрение всех узлов сети и наработку ими данной репутации r.

Для злоумышленника равновесное значение вероятности q, с которой он выбирает стратегию $m = 0$, определяется соотношением

$$q = 1 - \frac{N+M-1}{M\left(\frac{F}{N+M} + s(r)\right)}.$$

Отметим, что если выполнено неравенство (4.1) (низкая репутация), то формально $q < 0$ (игра решается в чистых стратегиях). В самом деле, равновесие в смешанных стратегиях означает, что при выигрыш игрока в среднем не зависит от его действий (при условии, что противник следует оптимальной стратегии). Если репутация низка, то вероятность p мала — сервер склонен к расточительной массовой

проверке ответов. Такой выбор обеспечивает в среднем нулевой выигрыш злоумышленника. Однако, если выполнено неравенство (4.1), то злоумышленник уже не может выбором $q \in [0,1]$ сравнять выигрыши сервера при различном выборе его действий: стратегия $p = 1$ доминирует. Возможна типичная для игровых моделей разрывность, если репутация близка к критической (при которой (4.1) обращается в равенство): для близких по репутации s узлов стратегии могут радикально отличаться: $p \approx 0$ при бóльших s и $p = 1$ при меньших. Практически это означает отказ от использования узлов с репутацией ниже критической.

Средние расходы сервера равны

$$-E(C) = (N+M) - s(r)\frac{N+M-1}{\frac{F}{N+M} + s(r)} = \frac{s(r)+F}{s(r)+\frac{F}{N+M}} > 1.$$

Единичный уровень означает безопасный расчет задания.

Рассмотрим несколько предельных случаев.

- Низкий штраф: $F \to 0$; в этом случае $-E(C) \to 1$, то есть расчет безопасен.
- Высокая репутация: $r \to \infty$; тогда тоже $-E(C) \to 1$ — расчет безопасен.
- В большой сети ($N \to \infty$, независимо от вторжения M) асимптотически достигается уровень расходов

$$-E(C) \to 1 + \frac{F}{s(r)}.$$

- Такой же асимптотический уровень расходов получается при массированном вторжении ($M \to \infty$).

Средний выигрыш злоумышленника $E(V^*)$ равен нулю.

Отметим, что оптимальные средние выигрыши не зависят от удельных расходов по наработке репутации I, от наживы злоумышленника G, и от числа внедренных узлов M (однако выигрыш сервера зависит от размера всей сети $N + M$, включая и внедренные узлы). Стратегия сервера p не зависит от неизвестного ему уровня внедрения M, то есть сервер фактически может принимать решения, основываясь на

полученных результатах (при условии, что ему известен доход от похищенного задания G и расходы I). Наконец, отметим, что стратегии зависят не от штрафа F и наживы G непосредственно, а от их удельных значений на каждый узел сети (включая и добросовестные), то есть от $F/(N+M)$ и $G/(N+M)$, соответственно.

Рассмотрим пример. Пусть $M = N = 10$, $G = 1$, $F = 10$, $I = 1$. Ограничение на репутацию: $s > 1.4$, примем $s = 2$. Пусть $s(r) = r$, то есть репутация — просто число выполненных заданий без разоблачения. Тогда сервер работает с узлами, которые выполнили хотя бы одно задание (первое — проверочное). Смешанные стратегии: $p = 0.975$, то есть сервер в 97% случаев не проверяет задания; $q = 0.24$ (злоумышленник в четверти случаев сотрудничает с проектом). Наконец, средние расходы сервера составляют 4.8. В случае проверки всех заданий расходы на задание равнялись бы в точности 20, а отказ от проверки приводил к бы к выплате штрафа в половине всех расчетов, что определило бы расходы в 6 единиц. Наконец, репликация на уровне 11 (если сервер угадал уровень вторжения) позволяет разоблачить все внедренные узлы и избежать штрафа, однако производительность все равно недопустимо низкая.

В случае более высокого штрафа $F = 100$ ограничение на s отсутствует (положим $s = 1$), и имеем $p = \frac{1}{1.05} \approx 0.95$, $q = \frac{4.1}{6} \approx 0.68$, $-E(C) = \frac{101}{6} \approx 16.8$. Репликация $\nu = 11$ дает в данном случае лучший результат, однако при репутации $s = 5$ это уже не так.

5. Ответы разной ценности

Задача с ответами разной ценности сводится к уже рассмотренной. Предположим, что решение каждого задачи (расчетного задания) может быть ценным, однако сравнительно редко, а может не представлять ценности, что случается значительно чаще. Такие задачи весьма распространены, в качестве примера можно указать поиск простых чисел (которых сравнительно мало) и упомянутый выше виртуальный скрининг. Обозначим вероятность обнаружения ценного ответа равна α. Получив ценный ответ, сервер проверяет его на доверенном компьютере: эти расходы в среднем равны $1 \cdot \alpha$ и, будучи константой, не влияют на выбор стратегии. Поэтому злоумышленнику имеет смысл скрыть только ценный ответ (и только утайка ценного ответа способна принести прибыль G). Поэтому в среднем

прибыль G заменяется на αG, что не меняет выводов по существу. Аналогично, сервер платит штраф только при утрате ценного ответа, что приводит к замене F на αF. Следует принять во внимание, что обнаружение ценного ответа приносит серверу некоторый доход $\bar{F}$. Возможны те же три случая:

1. Все реплики попали на узлы злоумышленника и неверный ответ принят сервером; доход злоумышленника и дополнительный расход сервера равны G и F, соответственно, если ответ был ценный, и нулю в противном случае.

2. Все реплики попали на честные узлы и принят правильный ответ; дополнительный расход сервера равен $-\bar{F}$, если ответ оказался ценным, и нулю, если нет; доход злоумышленника равен нулю в любом случае.

3. Все остальные распределения заданий по узлам, когда в расчете участвуют и честные, и нечестные узлы; дополнительный доход сервера $ms(r)$, а злоумышленник совершает эту работу, затрачивая $Ims(r)$ единиц стоимости. Этот доход сервера включает в себя и возможные премии за полученные в ходе наработки репутации ценные ответы. Однако если данный ответ оказался ценным, сервер получит свою премию $\bar{F}$.

Тогда средние расходы сервера и выигрыш злоумышленника, в среднем и с точностью до постоянного слагаемого, составляют

$$C = \nu + \frac{\alpha C_m^\nu}{C_{N+M}^\nu} F - \left(1 - \frac{C_m^\nu}{C_{N+M}^\nu}\right) \alpha \bar{F} - \left(1 - \frac{C_m^\nu}{C_{N+M}^\nu} - \frac{C_N^\nu}{C_{N+M}^\nu}\right) ms(r),$$

$$V = \frac{\alpha C_m^\nu}{C_{N+M}^\nu} G - \left(1 - \frac{C_m^\nu}{C_{N+M}^\nu} - \frac{C_N^\nu}{C_{N+M}^\nu}\right) Ims(r).$$

Константу $-1 \cdot \alpha \bar{F}$ в первом выражении можно отбросить, не меняя выводов по существу, а потому и матрица выигрышей (табл. 1) не меняется с учетом замен F на $(F + \bar{F})\alpha$ и G на αG.

6. Заключение

Применяя методы теории игр, удалось для простейшей модели сети добровольных вычислений с внедренными узлами злоумышленника получить следующие результаты:

- Расходы сервера, при оптимальных действиях злоумышленника, не меньше расходов в отсутствие злоумышленников, независимо от объема полезной работы, выполняемой ими для наработки репутации.

- Выигрыш злоумышленника, при оптимальных действиях сервера, отсутствует.

- Выигрыши (но не стратегии) от доходов и расходов злоумышленника не зависят.

- Стратегии игроков зависят от удельных значений штрафа и дохода от злонамеренных действий — в расчете на один узел (в том числе и честный).

- Стратегия сервера зависит только от доступных ему величин (в частности, от числа внедренных узлов она не зависит).

- Применение репутационной техники позволяет серверу терпеть низкие убытки, сравнимые со случаем знания уровня внедрения (позволяющим разоблачать злоумышленника ценой меньших затрат).

СПИСОК ЛИТЕРАТУРЫ

1. Ивашко Е.Е., Никитина Н.Н., Möller S. *Высокопроизводительный виртуальный скрининг в Enterprise Desktop Grid на базе BOINC* // Труды Международной суперкомпьютерной конференции «Научный сервис в сети Интернет». 2015. С. 58–60.

2. Sarmenta L.F.G. *Sabotage-tolerance mechanisms for volunteer computing systems* // Future Generation Computer Systems. 2002. V. 18. P. 561–572.

3. Khan M.K., Mahmood T., Hyder S.I. *Scheduling in Desktop Grid Systems: Theoretical Evaluation of Policies and Frameworks* //

International Journal of Advanced Computer Science and Applications. 2017. V. 8. N 1. P. 119–127.

4. Wang Y., Wei J., Ren Sh., Shen Yu. *Toward integrity assurance of outsourced computing a game theoretic perspective* // Future Generation Computer Systems. 2016. V. 55. P. 87–100.

5. Anta A.F., Georgiou Ch., Mosteiro M.A. Pareja D. *Multi-round Master-Worker Computing: A Repeated Game Approach* // IEEE 35th Symposium on Reliable Distributed Systems. 2016. P. 31–40.

6. Anta A.F., Georgiou C., Mosteiro M.A. Pareja D. *Algorithmic Mechanisms for Reliable Crowdsourcing Computation under Collusion* // PLoS ONE. V. 10. N 3. P. e0116520.

7. Christoforou E., Anta A.F., Georgiou Chryssis, Mosteiro M.A., Sánchez A. *Reputation-based mechanisms for evolutionary master-worker computing* // Principles of Distributed Systems. 2013. Lecture Notes in Computer Science. V. 8304. P. 98–113.

8. Mazalov V.V., Nikitina N.N., Ivashko E.E. *Hierarchical Two-Level Game Model for Tasks Scheduling in a Desktop Grid* // Applied Problems in Theory of Probabilities and Mathematical Statistics Related to Modeling of Information Systems. 2014. P. 641–645.

9. Nikitina N.N., Ivashko E.E., Tchernykh A. *Congestion Game Scheduling for Virtual Drug Screening Optimization* // Journal of Computer-Aided Molecular Design. 2018. V. 32. N 2. P. 363–374.

GAME-THEORETIC MODEL OF A VOLUNTEER COMPUTING GRID

Ilya A. Chernov, IAMR KarRC RAS, Cand.Sc.
(chernov@krc.karelia.ru).

Abstract: In the paper we propose a simple game-theoretic model of a Desktop Grid for volunteer computing. Task replication reduces the risk of accepting wrong answers due to sabotage. Saboteur's attack by intruding multiple computing nodes brings him some profit in case a wrong answer is accepted, while the server suffers some penalty in this case. Nodes are assigned some reputation as a monotone function of the number of produced correct (or not exposed) answers. We obtain the optimal mixed strategies and show that the average gain of the players depends only on the server's penalty, nodes' reputation, and the size of the subgrid of nodes with the same reputation. Also we estimate the server's cost per an answer. Numerical examples show that the average cost of the server is not more than that in the case when the number of intruders is known.

Keywords: volunteer computing, desktop grid, sabotage-tolerance, reputation.