

Алгоритм решения задачи обратного стресс-тестирования кредитного портфеля банка на основе системно-динамических моделей заемщиков

Д. С. Куренной

Аннотация—Статья посвящена алгоритму приближенного динамического программирования, решающему задачу обратного стресс-тестирования кредитного портфеля банка на основе системно-динамических моделей заемщиков. В алгоритме используются классические методы оптимизации квазиньютоновского типа.

Предложенная в статье реализация обратного стресс-тестирования рассматривает данную процедуру как способ построения сценариев факторов риска, приводящих к максимуму финансовых потерь кредитного портфеля и удовлетворяющих заданному критерию правдоподобия. При этом кредитный портфель формализован с помощью системно-динамических моделей, позволяющих описывать структуру и динамику сложных систем, которыми в контексте проведенного исследования являются предприятия различных отраслей российской экономики. В качестве факторов риска рассматриваются различные макроэкономические переменные, такие как мировые цены на нефть, курс доллара к рублю и др. Основой для построения сценариев факторов риска является многомерная ARIMA-GARCH-модель, обеспечивающая возможность учитывать корреляцию между различными макроэкономическими переменными.

Результаты работы метода приближенного динамического программирования сравниваются с результатами генетических алгоритмов. Доказанные в статье утверждения обосновывают корректность алгоритма и дают представление о его применимости к негладким задачам оптимизации, возникающим в контексте обратного стресс-тестирования и системной динамики. Основным инструментом реализации описываемого алгоритма является Matlab, обладающий широким спектром инструментов математического моделирования, в том числе графической средой имитационного моделирования Matlab Simulink, которая предназначена для построения системно-динамических моделей.

Ключевые слова—системная динамика, кредитные риски, обратное стресс-тестирование, имитационное моделирование, приближенное динамическое программирование, квазиньютоновские методы, Matlab, Simulink.

Статья получена 5 сентября 2018.

Д.С. Куренной, кафедра исследования операций факультета вычислительной математики и кибернетики МГУ им. М.В. Ломоносова, d.kurennoy@yandex.ru).

I. ВВЕДЕНИЕ

Одним из ключевых бизнес-процессов, определяющих эффективность деятельности банка, является управление кредитными рисками [6]. Большинство математических методов, предназначенных для оценки вероятности дефолта заемщика, сводятся к построению обобщенной рейтинговой модели на основе количественных показателей большой выборки данных аналогичных предприятий [21]. При этом указанные методы не учитывают структуру конкретных компаний, их динамику в условиях непрерывно меняющихся внешних факторов. Таким образом, распространенные методы не приспособлены для решения задач обратного стресс-тестирования [3, 5,7], которое в настоящее время рекомендуется регуляторами банковской сферы [11, 22, 23] в качестве одной из главных процедур оценки рисков. Обратное стресс-тестирование заключается в построении наиболее реалистичных сценариев, приводящих к заданному уровню финансовых потерь, или формировании достаточно реалистичных сценариев, максимизирующих потери банка. Результаты обратного стресс-тестирования могут быть использованы, в частности, для смягчения последствий реализации определенных сценариев.

Избавиться от ограниченности традиционных подходов позволяет применение для моделирования дефолтов системно-динамических моделей заемщиков [18, 19, 20]. Системная динамика [4, 14] обеспечивает возможность воспроизведения структуры исследуемого предприятия в виде непрерывно взаимодействующих элементов и внешних факторов. Связи между элементами описываются функциональными зависимостями и дифференциальными уравнениями, которые определяют динамику компании и степень ее устойчивости по отношению к различным макроэкономическим сценариям.

Данная статья посвящена описанию алгоритма, решающего задачу обратного стресс-тестирования кредитного портфеля банка, который представляется при помощи системно-динамических моделей заемщиков. Алгоритм основан на методе приближенного динамического программирования [10, 15], каждый шаг которого сводится к решению гладких

задач условной оптимизации при помощи метода последовательного квадратичного программирования [17]. В работе приведены утверждения, обосновывающие корректность сформулированного метода и дающие представление о его применимости к негладким задачам оптимизации, возникающим в процедуре обратного стресс-тестирования.

Статья имеет следующую структуру. Первый раздел посвящен постановке задачи обратного стресс-тестирования и описанию общих принципов использования системной динамики для моделирования дефолта компаний. Во второй части приводятся описание алгоритма решения задачи обратного стресс-тестирования и обоснование его корректности. Сформулированные утверждения позволяют сделать выводы о свойствах и применимости данного метода. В третьем разделе статьи представлено описание программного комплекса, с помощью которого реализован указанный алгоритм. Четвертая часть содержит анализ численных результатов и сравнение их с результатами генетического алгоритма.

II. ОБРАТНОЕ СТРЕСС-ТЕСТИРОВАНИЕ КРЕДИТНОГО ПОРТФЕЛЯ БАНКА

A. Постановка задачи обратного стресс-тестирования

Процедура стресс-тестирования активно используется в экономике и банковской сфере с конца прошлого века, однако финансовый кризис 2008 года показал необходимость модернизации сформировавшихся подходов к оценке рисков. В настоящий момент актуальной задачей, которую ставят перед банками регуляторы, является обратное стресс-тестирование, формализуемое двумя разными способами. Первый подход предполагает построение наиболее реалистичных сценариев, приводящих к заданному уровню финансовых потерь. Второй способ заключается в определении стресс-сценариев, максимизирующих финансовые потери и удовлетворяющих определенному критерию правдоподобия. Данная работа посвящена описанию метода решения задачи обратного стресс-тестирования следующего вида:

$$\max_{x \in R^{N \times T}} f(x) \quad (1)$$

$$L(x) \geq l \quad (2)$$

где x – вектор сценариев факторов риска; N – число факторов риска; T – количество кварталов, определяющих временной период, на котором осуществляется построение стресс-сценариев; $f(x)$ – функция финансовых потерь кредитного портфеля, возникающих вследствие дефолтов заемщиков; $L(x)$ – функция правдоподобия; l – константа, определяющая уровень правдоподобия искомых сценариев. При этом на целевую функцию (1) и функцию ограничений (2) накладываются условия гладкости, выпуклости и т.д.,

что приводит к необходимости решения, вообще говоря, негладкой задачи условной оптимизации.

В рамках данной работы в качестве факторов риска рассматриваются макроэкономические переменные (Табл. 1), динамика которых представляется при помощи многомерной ARIMA-GARCH-модели [1, 2, 16, 19]:

$$(1-V)^d X_t = c + \left(\sum_{i=1}^p a_i \cdot V^i \right) (1-V)^d X_t + \left(1 + \sum_{i=1}^q b_i \cdot V^i \right) e_t, \quad (3)$$

$$e_t = H_t^{1/2} \cdot \varepsilon_t. \quad (4)$$

Здесь $X_t = (x_t^1, \dots, x_t^n)^T$ – значение вектора макропараметров в квартале с номером t , $V: VX_t = X_{t-1}$ – лаговый оператор, a_1, \dots, a_p и b_1, \dots, b_q – векторы действительных чисел, являющихся коэффициентами авторегрессии и скользящего среднего соответственно, p, q и d – натуральные числа, определяющие порядок модели, $\{\varepsilon_t\}$ – вектор случайных процессов независимых одинаково распределенных по нормальному закону случайных величин, c – константа, H_t – матрица, которая может быть представлена в следующем виде:

$$H_t = C^{*T} C^* + \sum_{k=1}^r A_k^{*T} e_{t-1} e_{t-1}^T A_k^* + \sum_{k=1}^r G_k^{*T} H_{t-1} G_k^*, \quad (5)$$

где C^*, A_k^*, G_k^* – матрицы параметров размерностью $N \times N$.

Представление полного вида модели, включающего значения всех оцененных коэффициентов, в рамках данной статьи не представляется возможным, однако производится описание результатов оценки ее адекватности. Калибровка ARIMA-GARCH-модели производилась на основе исторических квартальных данных периода 2004-2014 гг. Значения параметров ($p = 3, d = 2, q = 2, r = 1$) обусловлены минимальной величиной Байесовского информационного критерия ($BIC = 2353,2$) [12]. Статистическая значимость и качество модели оценены при помощи асимптотического теста Фишера и анализа стандартных ошибок коэффициентов авторегрессий. Ошибки аппроксимации полученных макроэкономических переменных, рассчитанные по формуле (6), не превышают 9,8%.

$$\overline{err}_i = \frac{1}{n_{hist}} \sum_{t=1}^T \left| \frac{x_t^i - \tilde{x}_t^i}{x_t^i} \right| \cdot 100\%, i = 1, \dots, N, \quad (6)$$

где n_{hist} – объем выборки ($n_{hist} = 40$), \tilde{x}_t^i – исторические значения i -ой макроэкономической переменной, x_t^i – значения, полученные моделью ARIMA-GARCH для i -ого макропараметра.

При заданных коэффициентах ARIMA-GARCH-модели поведение внешних факторов определяется случайными величинами e_t , функция правдоподобия которых для случая нормального распределения записывается следующим образом:

$$\prod_{t=2}^T \frac{1}{(2\pi)^{k/2} |H_t|^{1/2}} e^{-\frac{1}{2}(e_t)^T H_t^{-1} e_t}.$$

В результате задача (1)-(2) принимает вид:

$$\max_{e \in \{e_1, \dots, e_T\}} f(e) \quad (7)$$

$$\prod_{t=2}^T \frac{1}{(2\pi)^{k/2} |H_t|^{1/2}} e^{-\frac{1}{2}(e_t)^T H_t^{-1} e_t} \geq l. \quad (8)$$

В рамках процедуры приближенного динамического программирования критерий (7) заменяется критерием минимума (9) выражения, представляющего взвешенную сумму средств, которыми располагают компании. При этом весами являются соответствующие объемы выданных кредитов:

$$\min_{e \in \{e_1, \dots, e_T\}} (D_t^T \otimes S^T) \cdot R_t, \quad (9)$$

где \otimes - прямое (покомпонентное) произведение векторов-строк, S - вектор сумм выданных кредитов (постоянный по времени); D_t - вектор, i -я компонента которого принимает значение 0 в случае дефолта i -ого заемщика в любом из предыдущих моментов времени, иначе 1; R_t - вектор, характеризующий количество средств, которыми располагает каждая компания. Величина R_t определяется запуском системно-динамических моделей. Равенство нулю i -й компоненты вектора R_t означает дефолт i -ого заемщика.

В. Системная динамика в процедуре обратного стресс-тестирования

Рассматриваемый в статье подход к решению задачи обратного стресс-тестирования основан на системно-динамических моделях заемщиков. Системная динамика представляет собой подход имитационного моделирования, предназначенный для описания структуры и динамики сложных систем на основе концепции потоков, накопителей и обратных связей, которым соответствует формальное представление в виде системы дифференциальных уравнений. На основе финансовой отчетности периода 2007-2015 гг. и информации из других открытых источников были разработаны системно-динамические модели предприятий различных отраслей российской экономики (Табл. 1): нефтедобывающей и нефтеперерабатывающей компании, сельскохозяйственного холдинга, производителей алюминиевого профиля и микроэлектроники, дистрибьютора алкогольной продукции, официального дилера российских и зарубежных производителей автомобилей. Подробное описание системно-динамических моделей и принципов их построения представлено в [19, 20].

В указанных моделях можно выделить две смысловые компоненты, непрерывно взаимодействующие между собой и соответствующие различным видам деятельности компаний: производственную и финансовую. Основные накопители производственной части агрегируют материальные

факторы, являющиеся сырьевыми ресурсами или готовой для продажи продукцией конкретного предприятия. Потоки обеспечивают взаимосвязь данных накопителей и реализуют, таким образом, производственный процесс. Одним из накопителей финансовой части является накопитель «Долг», который отражает текущий объем долговых обязательств компании. Остальные накопители и потоки данной части позволяют отразить особенности кредитной политики предприятия.

Ключевыми элементами всех системно-динамических моделей заемщиков, обобщающими деятельность компании и объединяющими финансовую и производственную части, являются накопители «Средства в рублях», значения которых образуют вектор R_t . Равенство нулю i -ого накопителя «Средства в рублях» означает дефолт i -ого предприятия.

Совокупность данных моделей позволяет сформировать пример кредитного портфеля банка и связать уровень его потерь со сценариями макроэкономических параметров, обуславливающими дефолт рассматриваемых предприятий. В качестве основных макропараметров, которые представлены в моделях независимыми переменными, выступают мировые цены на нефть, курс доллара к рублю, ставка привлекаемых и погашаемых кредитов и цены на продукцию, которой торгуют компании (Табл. 1).

Табл. 1. Заемщики и рассматриваемые макроэкономические переменные.

Название компании	Сайт компании	Рассматриваемые макроэкономические переменные.
«АРОМА»	http://aroma.ru/	- курс доллара к рублю; - ставка привлекаемых, погашаемых кредитов (Mosprime); - цены на шампанское, вино, крепкий алкоголь дорогих и обычных сортов на внутреннем и внешнем рынке.
«ВМК»	http://vmk-etm.ru/	- мировые цены на нефть; - курс доллара к рублю; - ставка привлекаемых, погашаемых кредитов (Mosprime); - мировые цены на алюминий; - цены алюминиевого профиля: экспортная и российская.
«Ангстрем»	https://www.angstrem.ru/ru	- мировые цены на нефть; - курс доллара к рублю; - ставка привлекаемых, погашаемых кредитов (Mosprime); - цены на кремний в России.
«У Сервис+»	http://www.uservice.ru/	- мировые цены на нефть; - курс доллара к рублю; - ставка привлекаемых, погашаемых кредитов (Mosprime); - закупочные цены автомобилей российских и иностранных марок экономического и бизнес классов; - цены автомобилей на

		внутреннем рынке российских и иностранных марок экономического и бизнес классов.
«Агросила»	http://agrosila-holding.ru/	- мировые цены на нефть; - курс доллара к рублю; - ставка привлекаемых, погашаемых кредитов (Mosprime); - цены на бензин на внутреннем рынке; - цены на продукцию которой торгует компания: сахар, семена, корм, мясо, зерновые, молоко, яйца, куриное мясо.
«Башнефть»	http://bashneft.ru/	- экспортные и внутренние цены на нефть, дизель, мазут, бензин; - курс доллара к рублю; - ставка привлекаемых, погашаемых кредитов (Mosprime); - удельные себестоимости добычи и переработки нефти; - базовая ставка НДС.

каждой пары состояний: текущего этапа (в момент времени t) и следующего за ним этапа (в момент времени $t+1$). Состояния соответствуют различным ограничениям на текущее значение функции правдоподобия при решении указанных подзадач. В результате на каждом этапе t определяются состояния, переход из которых в каждое новое состояние следующего этапа $t+1$ дает наилучшее значение критерия оптимизации. Таким парам состояний соответствуют изменения макроэкономических параметров, обеспечивающие переход между ними. Общая схема одного этапа алгоритма иллюстрируется на Рис. 1. Черные стрелки указывают на пары состояний найденного оптимального перехода из состояния в момент времени t в состояние шага $t+1$, пунктиром отображаются все рассматриваемые пары.

На шаге T определяется конечное состояние (m^*), соответствующее минимуму целевой функции исходной задачи (8)-(9) и соответствующие значения $e_T^{m^*}$.

III. АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ ОБРАТНОГО СТРЕСС-ТЕСТИРОВАНИЯ

A. Общая схема алгоритма динамического программирования

В рамках данной работы для решения задачи (8)-(9) применяется метод приближенного динамического программирования [10, 15], основанный на принципе «движения вперед». Процесс разработки и формализации указанного алгоритма можно разбить на несколько пунктов в соответствии с общими принципами динамического программирования:

Выбор способа разбиения исходной задачи на этапы.

Формализация понятия состояний, рассматриваемых на каждом этапе. При этом каждому состоянию должны соответствовать конкретные значения параметров задачи и искомым переменных.

Обоснование способа определения оптимального перехода из одного состояния в другое. Постановка оптимизационных задач, решаемых для этого на каждом этапе.

Определение способа получения решения динамической задачи на основе информации о состояниях и оптимальных переходах между ними.

В данном случае процедура построения стресс-сценариев разбивается на T этапов в соответствии с количеством кварталов, на которых исследуется изменение макроэкономических переменных. В качестве основной характеристики состояний рассматривается величина l , задающая ограничения на текущее значение логарифмической функции правдоподобия, фигурирующей в (8). Таким образом, состояние определяет точный вид ограничений на величину функции правдоподобия в конкретный момент времени.

Для определения оптимального перехода между состояниями решаются оптимизационные подзадачи для

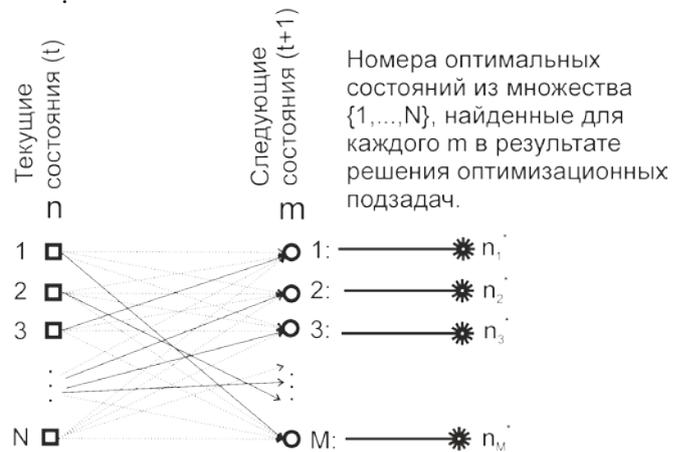


Рис. 1. Общая схема одного шага алгоритма приближенного динамического программирования.

Применяемая после этого процедура обратного хода алгоритма позволяет определить искомый сценарий макроэкономических переменных с учетом полученной в процессе прямого хода информации об оптимальных переходах между состояниями. Так, зная номер (n_m^*) состояния момента времени $T-1$, обеспечивающего оптимальный переход в состояние m^* , можно определить соответствующие значения величин $e_{T-1}^{n_m^*}$. Данная процедура повторяется до начального момента времени. Полученная последовательность величин $e_t, t = 1, \dots, T$ с учетом (3) и (4) задает искомый стресс-сценарий.

B. Формальное описание алгоритма решения задачи обратного стресс-тестирования

В данном разделе представлено формальное описание алгоритма решения задачи (8)-(9) и ряд утверждений, обосновывающих некоторые из его этапов.

Для реализации процедуры динамической оптимизации в каждый из будущих моментов времени рассматривается множество состояний факторов

неопределенности (в дальнейшем ФН). Данные состояния получаются в результате использования различных ограничений на текущее значение логарифмической функции правдоподобия при решении оптимизационных задач, соответствующих рассматриваемому моменту времени. Введем следующие обозначения:

$t = 1, \dots, T$ – рассматриваемые в моделях заемщиков моменты времени. $t = 1$ соответствует начальному моменту;

k – число заемщиков в портфеле;

N_t – число рассматриваемых состояний ФН в момент времени t (в начальный момент времени состояние одно);

D_t^n – вектор дефолтности заемщиков в момент времени t в состоянии ФН n , $n = 1, \dots, N_t$. Размерность вектора равна K , элементы вектора равны 1 для недефолтных заемщиков и равны 0 для заемщиков, находящихся в дефолте;

D_{t+1}^{nm} – вектор дефолтности заемщиков в момент времени $t+1$ в состоянии ФН m при условии, что в момент t ФН находились в состоянии n ;

S – вектор сумм кредитов заемщикам (размерности K);

R_{t+1}^{nm} – вектор состояний накопителей «средства в рублях» всех заемщиков в момент времени $t+1$ в состоянии ФН m , если в момент t ФН находились в состоянии n ; данный вектор определяется в результате запуска системно-динамических моделей.

A_t^n – вектор состояний остальных накопителей системно-динамической модели в момент t , если ФН находятся в состоянии; данный вектор определяется в результате запуска системно-динамических моделей.

Векторы A_1^1 и R_1^1 известны в момент решения оптимизационной задачи – это начальные состояния накопителей системно-динамических моделей;

e_t^{nm} – вектор ошибок многомерной ARIMA-GARCH модели, соответствующий переходу ФН из состояния n в момент времени t в состояние m в момент времени $t+1$;

H_t^n – матрица ковариаций ошибок многомерной ARIMA-GARCH модели, соответствующая состоянию ФН n в момент времени t . Матрица H_1^1 известна в момент решения оптимизационной задачи;

G_t^n – значение логарифмической функции правдоподобия в момент времени t в состоянии ФН n ;

$M_t^n, t = 2, \dots, T$ – номер состояния ФН в момент времени $t-1$, переход из которого в состояние n в следующий момент времени t дает наилучшее значение критерия оптимизации.

Ограничение на величину функции правдоподобия имеет вид:

$$\prod_{t=2}^T \frac{1}{(2\pi)^{k/2} |\mathbf{H}_t|^{1/2}} e^{-\frac{1}{2}(e_t)^\top \mathbf{H}_t^{-1} e_t} \geq e^{\underline{G}}, \quad (10)$$

при этом

$$\mathbf{H}_t = \mathbf{H}_t(\mathbf{H}_{t-1}, e_{t-1}), t = 2, \dots, T. \quad (11)$$

Логарифмируя обе части неравенства (10), получим:

$$-\frac{k(T-1)}{2} \ln(2\pi) - \sum_{t=2}^T \ln \left(|\mathbf{H}_t|^{1/2} \right) - \frac{1}{2} \sum_{t=2}^T (e_t)^\top \mathbf{H}_t^{-1} e_t \geq \underline{G}. \quad (12)$$

Таким образом, минимальное значение логарифмической функции правдоподобия равно \underline{G} . Максимальное значение данной функции может быть найдено в результате решения оптимизационной задачи

$$\max_{e_t} -\frac{k(T-1)}{2} \ln(2\pi) - \sum_{t=2}^T \ln \left(|\mathbf{H}_t|^{1/2} \right) - \frac{1}{2} \sum_{t=2}^T (e_t)^\top \mathbf{H}_t^{-1} e_t, \quad (13)$$

с ограничением (12). Решение задачи (13), (12) обозначим \bar{G} . Состоянию ФН n , $n = 1, \dots, N_t$ в момент времени t будет соответствовать значение функции правдоподобия $\tilde{G}_t^n = \underline{G} + \frac{n-1}{N_t-1} (\bar{G} - \underline{G})$.

Алгоритм

А. Получить значение \bar{G} путем решения задачи (13), (12) и значение \underline{G} . Рассчитать величины $\tilde{G}_t^n, t = 2, \dots, T, n = 1, \dots, N_t$.

В. $t := 1$.

С. $D_t^n := 1, L_t^n := 0, n = 1, \dots, N_t$.

Д. Решить оптимизационные задачи

$$F_{t+1}^{nm} := \min_{e_t^{nm}} [(D_t^{n\top} \otimes S^\top) \cdot R_{t+1}^{nm}], \quad (14)$$

$$G_t^n - \frac{k}{2} \ln(2\pi) - \ln \left(|\mathbf{H}_t^n|^{1/2} \right) - \frac{1}{2} (e_t^{nm})^\top (\mathbf{H}_t^n)^{-1} e_t^{nm} \geq \tilde{G}_{t+1}^m. \quad (15)$$

$$n = 1, \dots, N_t, m = 1, \dots, N_{t+1}.$$

Если какая-либо из задач (14), (15) не имеет решения, то переход из n в m исключается.

Е. По результатам п.Д по состояниям системно-динамических моделей для каждого m найти

$$n^* = \arg \min_n S^\top D_{t+1}^{nm}. \quad (16)$$

Если существует несколько различных значений n , удовлетворяющих условию (16) то выбрать в качестве n^* то из них, для которого меньше величина F_{t+1}^{nm} .

Присвоить

$$\begin{aligned} M_{t+1}^m &= n^* , \\ D_{t+1}^m &:= D_{t+1}^{n^*} , \\ R_{t+1}^m &:= R_{t+1}^{n^*} , \\ A_{t+1}^m &:= A_{t+1}^{n^*} , \\ G_{t+1}^m &:= G_t^{n^*} - \frac{k}{2} \ln(2\pi) - \\ & - \ln \left(\left| \mathbf{H}_t^{n^*} \right|^{\frac{1}{2}} \right) - \frac{1}{2} \left(\mathbf{e}_t^{n^*} \right)^T \left(\mathbf{H}_t^{n^*} \right)^{-1} \mathbf{e}_t^{n^*} \end{aligned} \quad (17)$$

F. $t := t + 1$. Если $t < T$, то идти к п. D.

G. Найти $m^* = \arg \min_m S^T D_t^m$. Заданная в обратном порядке последовательность состояний m^* , $M_T^{m^*}, M_{T-1}^{m^*}, \dots, 1$ и соответствующие значения инноваций $\mathbf{e}_t^{n^*, m^*}, t = 1, \dots, T - 1$ определяют сценарий обратного стресс-тестирования.
H. Конец.

На каждом шаге общего описанного алгоритма рассматривается совокупность задач оптимизации следующего вида:

$$\begin{aligned} \min_{\mathbf{e}_t^{nm}} & \left[\left(D_t^{n^*} \otimes S^T \right) \cdot R_{t+1}^{nm} \right], \quad (18) \\ G_t^{n^*} - \frac{k}{2} \ln(2\pi) - \ln \left(\left| \mathbf{H}_t^{n^*} \right|^{\frac{1}{2}} \right) - \\ & - \frac{1}{2} \left(\mathbf{e}_t^{nm} \right)^T \left(\mathbf{H}_t^{n^*} \right)^{-1} \mathbf{e}_t^{nm} \geq \tilde{G}_{t+1}^{n^*}. \end{aligned} \quad (19)$$

При этом величине располагаемых заемщиками средств $R_{t+1}^{nm}(\mathbf{e}_1^{1,2}, \dots, \mathbf{e}_t^{nm})$, которая зависит от макроэкономических переменных, определяемых случайными величинами e_t , соответствует представление, основанное на структуре входящих $(f_R^+(R, t, e), f_A^+(A, R, t, e))$ и исходящих $(f_R^-(R, t, e), f_A^-(A, R, t, e))$ потоков системно-динамических моделей, которые могут включать в себя как положительные, так и отрицательные обратные связи [4, 14]:

$$\begin{aligned} \frac{dR(t, e)}{dt} &= f_R^+(R, t, e) - f_R^-(R, t, e) \Leftrightarrow \\ \Leftrightarrow R(t, e) &= \int_0^t [f_R^+(R, \tau, e) - f_R^-(R, \tau, e)] d\tau, \end{aligned} \quad (20)$$

$$\begin{aligned} \frac{de}{dt} &= \frac{e_t - e_{t-1}}{t - (t-1)} = \\ &= H_t^{1/2} \cdot \varepsilon_t - H_{t-1}^{1/2} \cdot \varepsilon_{t-1} = f_e(t, e). \end{aligned} \quad (21)$$

$$\begin{aligned} \frac{dA(R, t, e)}{dt} &= f_A^+(A, R, t, e) - f_A^-(A, R, t, e) \Leftrightarrow \\ \Leftrightarrow A(t, e) &= \int_0^t f_A^+(A, R, \tau, e) - f_A^-(A, R, \tau, e) d\tau \\ R(t_0, e) &= R_0, A(R_0, t_0, e) = A_0. \end{aligned} \quad (22)$$

При этом $\varepsilon_t, \varepsilon_{t-1}$ – константы.

При непрерывности $f_R^+(R, t, e), f_R^-(R, t, e), f_e(t, e)$ по t и непрерывности частных производных функций $f_R^+(R, t, e), f_R^-(R, t, e)$ по R задача (20), (23) имеет единственное решение (по теореме Коши).

Утверждение 1. Зависимости $f_R^+(R, t, e), f_R^-(R, t, e)$ являются непрерывно дифференцируемыми по $\mathbf{e} \in \mathbb{R}^n$.

Доказательство: Из описанных общих принципов построения системно-динамических моделей [4, 14, 19, 20] следует, что данные функции представляют собой сумму входящих и исходящих потоков накопителя «Средства в рублях», каждый из которых является суперпозицией нескольких функциональных зависимостей, объединяющих воздействующие на указанные потоки элементы модели:

$$\begin{aligned} f_R^+(R, t, e) &= Q_1(f_1^+(R, t, e), \dots, f_{q_1}^+(R, t, e)), \\ f_R^-(R, t, e) &= Q_2(f_1^-(R, t, e), \dots, f_{q_2}^-(R, t, e)). \end{aligned}$$

В данных формулах, вообще говоря, не все функции могут зависеть от полного набора переменных. При этом, $Q_1(\cdot), Q_2(\cdot)$ представляют собой комбинации операций сложения и вычитания и не влияют на свойство дифференцируемости, следовательно, гладкость указанных в условии функций определяется только дифференцируемостью $f_1^+(R, t, e), \dots, f_{q_1}^+(R, t, e)$ и $f_1^-(R, t, e), \dots, f_{q_2}^-(R, t, e)$. Таким образом, чтобы доказать утверждение, необходимо рассмотреть все типы функций, используемых при построении системно-динамических моделей и обосновать их гладкость относительно \mathbf{e} .

1. Величина \mathbf{e} вводится при описании ARIMA-GARCH-моделей, соответствующих макропараметрам и

определяющих линейную зависимость макропараметров от e . Отсюда следует непрерывная дифференцируемость функций, соответствующих макропараметрам, от e .

2. Типы функций, выражающих зависимости остальных элементов модели от функций макропараметров, следующие:

- а) полиномиальные зависимости различного (ненулевого) порядка, полученные в результате регрессионного анализа;
- б) обратные зависимости;
- в) сплайн-функции¹.

Очевидно, что в случаях а) и в) дифференцируемость сохраняется с учетом 1. Гладкость в случае б) обеспечивается за счет того, что рассматриваемые макропараметры не принимают нулевых значений. Утверждение доказано.

Утверждение 2. Функция $f_e(t, e)$, описываемая формулой (20), является непрерывно дифференцируемой по $e \in R^N$.

Доказательство: Исходная функция представляет собой производную e по t . Доказательство напрямую следует из формулы для e и **Н**.

Данные утверждения позволяют обосновать применимость квазиньютоновских методов к задаче (18)–(19) и сделать выводы о свойствах каждого шага алгоритма приближенного динамического программирования.

Решения задачи (18)–(19) осуществляется классическим методом последовательного квадратичного программирования (SQP; от английского «Sequential Quadratic Programming»; см. [17]) с использованием механизма точных штрафных функций для его глобализации. Опишем данный алгоритм для оптимизационной задачи с ограничениями – неравенствами:

$$\min_{x \in R^N} F(x) \tag{24}$$

$$c(x) \geq 0, \tag{25}$$

где $f: R^N \rightarrow R$ и $c: R^N \rightarrow R^{N^0}$ – гладкие функции.

Функция Лагранжа задачи (24)–(25) выглядит следующим образом

$$L(x, \lambda) = F(x) + \lambda^T c(x), \tag{26}$$

Метод SQP определяет соответствующий сдвиг d_k из точки x_k как решение следующей задачи квадратичного программирования, аппроксимирующей исходную задачу (24)–(25):

$$\min_{d \in R^N} g(x_k)^T d + \frac{1}{2} d^T W(x_k, \lambda_k) d, \tag{27}$$

$$c(x_k) + A(x_k) d \geq 0, \tag{28}$$

где $g(x) = \nabla F(x)$, $A(x)$ – матрица Якоби ограничений $c(x)$, а $\lambda \in R^{N^0}$ – множители Лагранжа.

При этом

$$W(x, \lambda) \approx \nabla_{xx}^2 L(x, \lambda) = \nabla_{xx}^2 f(x) + \sum_{i=1}^{N^0} \lambda^i \nabla_{xx}^2 c^i(x) \tag{29}$$

– аппроксимация матрицы Гессе функции Лагранжа по методу Бройдена-Флетчера- Гольдфарба-Шанно (BFGS) [16]. Здесь $c^i(x)$ обозначает i -ую функцию из ограничений задачи, λ^i – соответствующий множитель Лагранжа. Новая итерация будет определяться следующим образом:

$$x_{k+1} = x_k + \alpha_k d_k,$$

где $\alpha_k \in (0, 1]$ – параметр длины шага, подбираемый на основе правила Армихо [17] для штрафной функции $\varphi_\pi(x) = f(x) + \pi \|c(x)\|$:

$$\varphi_\pi(x_k + \alpha_k d_k) \leq \varphi_\pi(x_k) + \eta \alpha_k D\varphi_\pi(d_k), \tag{30}$$

где $D\varphi_\pi(d_k) = g_k^T d_k - \pi \|c_k\|$ – производная по направлению функции φ_π вблизи d_k , $0 < \eta < 1$.

Алгоритм последовательного квадратичного программирования:

Фиксируем $x_0, \lambda_0, \{\pi_k\} \rightarrow \infty, 0 < \eta < 1$. Для $k = 0, \dots$ пока условие остановки метода не удовлетворено, выполняем следующие шаги:

1. Вычисляем f_k, g_k, c_k, W_k, A_k , берем $\alpha_k = 1$.
2. Ищем (d_k, δ_k) из (27)–(28).
3. Пока не выполнится правило Армихо,

уменьшаем длину шага: $\alpha_k = \frac{\alpha_k}{2}$.

4. В качестве нового приближения возьмем $(x_{k+1}, \lambda_{k+1}) = (x_k, \lambda_k) + \alpha_k (d_k, \delta_k)$.

Утверждение 3. Пусть $f_e(t, e) \neq 0$ на множестве $[0, T] \times R^N$. Тогда целевая функция и функция ограничений задачи (18)–(19) дважды непрерывно дифференцируемы по $e \in R^N$.

Доказательство: Очевидно, что гладкость целевой функции задачи (18)–(19) зависит от дифференцируемости R . При этом:

$$\frac{dR(e)}{de} = \frac{dR}{dt} \cdot \frac{dt}{de} = \frac{f_R^+(R, t, e) - f_R^-(R, t, e)}{f_e(t, e)}. \tag{31}$$

По условию $f_e(t, e) \neq 0$ на множестве $[0, T] \times R^N$, а функции $f_e(t, e), f_R^+(R, t, e),$

¹ В построенных моделях, приведенных в [19, 20] присутствуют кусочно-линейные функции, которые в рамках данного исследования были заменены сплайнами.

$f_R^-(R, t, e)$ непрерывно дифференцируемы согласно утверждениям 1 и 2. Отсюда следует, что $R(e)$ дважды непрерывно дифференцируема:

$$\frac{d^2 R(e)}{d^2 e} = \frac{\frac{\partial [f_R^+(R, t, e) - f_R^-(R, t, e)]}{\partial e} \cdot f_e(t, e)}{[f_e(t, e)]^2} \quad (32)$$

$$= \frac{[f_R^+(R, t, e) - f_R^-(R, t, e)] \cdot \frac{\partial [f_e(t, e)]}{\partial e}}{[f_e(t, e)]^2}.$$

Гладкость ограничений следует из положительной определенности \mathbf{H} . Утверждение доказано.

Доказанные факты позволяют сформулировать утверждение, которое обосновывает применение в алгоритме динамического программирования квазиньютоновских методов и является прямым следствием этих результатов и теоремы о сходимости квазиньютоновских методов. [17].

Утверждение 4. Пусть $f_e(t, e) \neq 0$ на своей области определения, целевая функция и функция ограничений задачи (18)–(19) дважды непрерывно дифференцируемы в некоторой окрестности точки \bar{e} , причем их вторые производные непрерывны в этой точке. Пусть в точке \bar{e} выполнено условие линейной независимости, причем \bar{e} – стационарная точка задачи (18)–(19), а $\bar{\lambda}$ однозначно отвечающий ей множитель Лагранжа. $(\bar{e}, \bar{\lambda})$ – стационарная точка задачи (18)–(19). Пусть в точке e выполнено достаточное условие второго порядка оптимальности, а также условие строгой дополнителности.

Тогда любое начальное приближение (e^0, λ^0) , достаточно близкое к $(\bar{e}, \bar{\lambda})$, корректно определяет траекторию алгоритма последовательного квадратичного программирования, применяемого для решения задачи (18)–(19). Скорость сходимости сверхлинейная, а если вторые производные целевой функции и ограничений непрерывны по Липшицу в окрестности $(\bar{x}, \bar{\lambda})$, то квадратичная.

Доказательство данного утверждения полностью повторяет доказательство в [17] с учетом условия $f_e(t, e) \neq 0$ и утверждений 1, 2, 3, которые утверждают гладкость целевой функции и функции ограничений до второго порядка включительно.

С. Пример неоптимальной работы алгоритма приближенного динамического программирования

В некоторых случаях описанный алгоритм приближенного динамического программирования не позволяет найти оптимальное решение задачи обратного стресс-тестирования. Одной из причин является способ

выбора n^* (представлен формулой (15)), согласно которому алгоритм отдает предпочтение состояниям, обеспечивающим дефолт компании. При этом может возникнуть ситуация, когда состояния, не приводящие к дефолту в текущем квартале гарантируют разорение предприятия с большим весом в более поздний момент времени, однако они будут отвергнуты на основе указанного принципа определения n^* .

Рассмотрим в качестве примера двух заемщиков с разными объемами кредитов и начальными состояниями накопителя «Средств в рублях» в течение четырех кварталов: $S = (30, 100)$, $R_0 = (12, 100)$, $T = 4$, значение целевой функции $F_0 = 10360$. Схема, иллюстрирующая процесс решения задачи методом приближенного динамического программирования представлена на Рис. 2. Последовательность состояний, выделенная алгоритмом, отмечена черными закрашенными кругами, соответствующие значения R и F записаны рядом с номером состояния. Ключевым в данном случае является поиск состояния момента времени $t=2$, обеспечивающего оптимальный переход в состояние $m=1$ момента времени $t=3$. Описанный алгоритм отдает предпочтение состоянию с номером $n=2$, так как переход из этого состояния обеспечивает дефолт первого заемщика. При этом оптимальным является состояние $n=1$, переход из которого не приводит к разорению компаний в момент времени $t=3$, но обуславливает дефолт второго заемщика в четвертом квартале. Отвергнутая оптимальная траектория для рассматриваемой задачи изображена на Рис. 3. Таким образом, значение целевой функции задачи $F = 270$ (Рис. 3) будет недостижимо, и алгоритм приближенного динамического программирования остановится на значении $F = 300$ (Рис. 2).

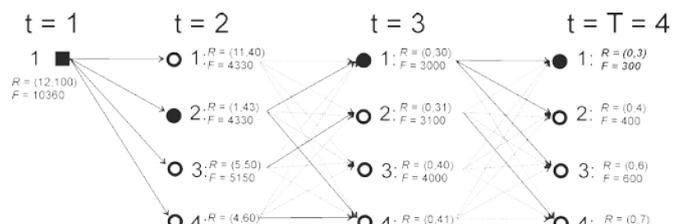


Рис. 2. Пример неоптимальной работы алгоритма приближенного динамического программирования.

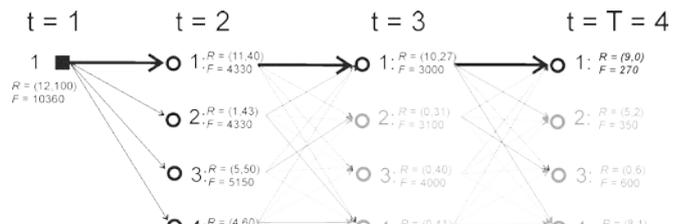


Рис. 3. Оптимальная траектория алгоритма, отвергнутая на основании условия (15).

программирования на каждом этапе представлены в Табл. 3. Таблица содержит итоговые номера оптимальных состояний, а также соответствующие им значения функции ограничений и целевой функции, уменьшение которой на каждом этапе обуславливается изменением величин R_t и D_t по времени.

Табл. 3. Траектория работы алгоритма приближенного динамического программирования.

Номер шага	Номер оптимального состояния	Значение функции ограничений	Значение целевой функции (млн. руб.)
1	1	-170,3	817600
2	25	-343,8	616800
3	24	-502	579900
4	22	-668,6	540300
5	24	-811,1	512700
6	23	-940,7	468800
7	24	-1061,4	422400
8	22	-1185,2	350460
9	25	-1292,8	269330
10	20	-1412	184100
11	21	-1514,5	151190
12	18	-1622,1	90260
13	15	-1732,4	65550
14	14	-1828,6	57815
15	11	-1940,9	39600
16	20	-2041,8	31740
17	23	-2150,9	7300
18	18	-2252,2	3770
19	14	-2361,5	2850
20	12	-2457,3	899
21	6	-2555,8	565
22	2	-2647,1	460

Построенные макроэкономические сценарии основных внешних параметров представлены на Рис. 5-Рис. 7. В данном случае для портфеля компаний разных отраслей российской экономики критичным является следующее поведение ключевых макропараметров: снижение мировых цен нефти до 50 долларов за баррель, стабилизация курса доллара в районе 67 рублей, установление ставки Mosprime на уровне 12%. В результате реализации такого макроэкономического сценария в течение пяти с половиной лет дефолт претерпевают четыре заемщика из шести с общей суммой кредитов 177 миллионов рублей ($D_{22} = (0, 0, 1, 1, 0, 0)^T$).

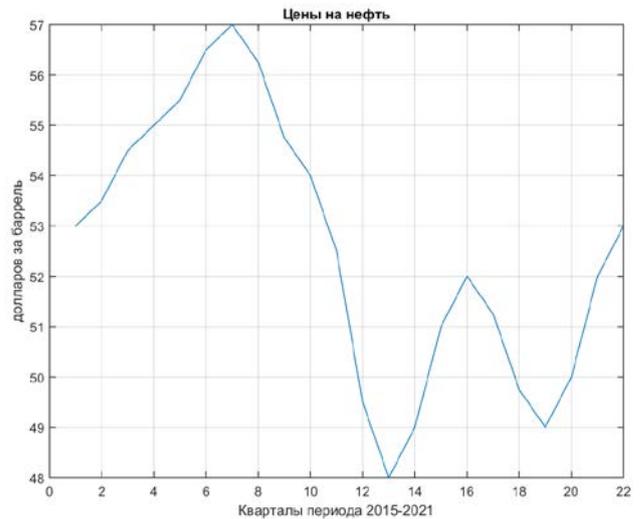


Рис. 5 Динамика мировых цен на нефть в найденном сценарии.

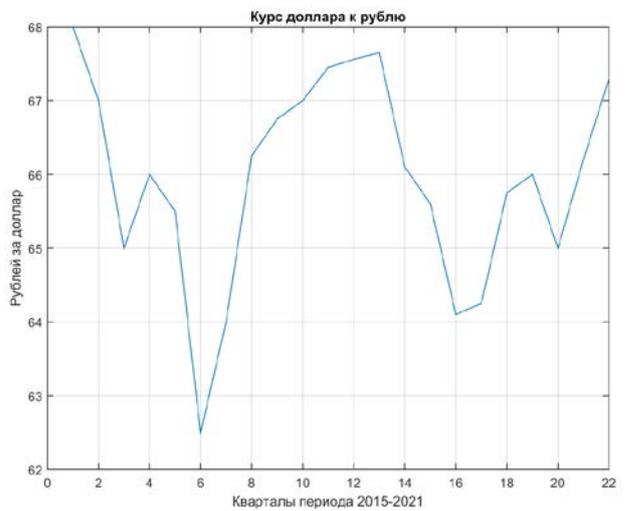


Рис. 6 Динамика курса доллара в найденном сценарии.

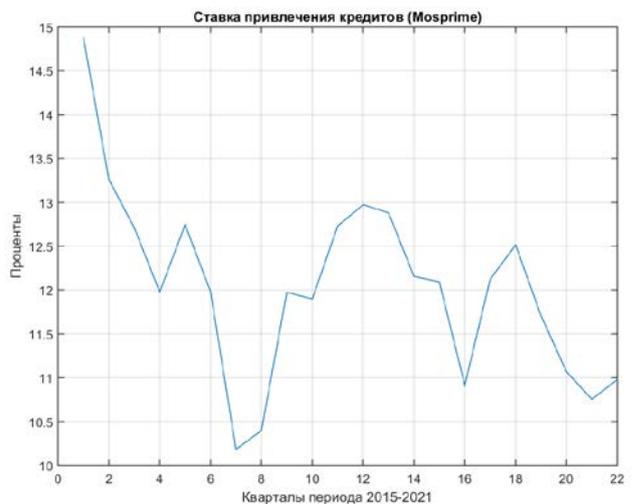


Рис. 7 Динамика ставки Mosprime в найденном сценарии.

В. Сравнение эффективности метода динамического программирования с результатами генетических алгоритмов

Генетические алгоритмы [13] основаны на принципе поиска решения оптимизационной задачи путем последовательного улучшения исходных величин искомых факторов (начальной популяции) с помощью применения к ним специальных операторов, обеспечивающих возможность комбинирования и подбора лучших с точки зрения критерия оптимизации значений.

Параметры различных запусков генетического алгоритма и метода приближенного динамического программирования представлены в Табл. 4 и Табл. 5. Начальная популяция для генетического алгоритма в данном случае задавалась ARIMA-GARCH-моделью, построение стресс-сценариев осуществлялось для временного периода 12 кварталов. Результаты работы методов представлены на Рис. 8 и Рис. 9.

Табл. 4. Параметры запусков генетического алгоритма и метода динамического программирования с разным количеством заемщиков.

Параметр	Значение параметра
Количество заемщиков (k) для разных запусков	[1;2;3;4;5;6]
Количество внешних параметров (N)	4
Временной период (T)	12 кварталов (3 года)
Вектор кредитов S , млн. руб. (для k -ого запуска рассматривались k первых компонент, $k = 1, \dots, 6$)	$(15, 18, 21, 25, 34, 100)^T$
Численность популяции генетического алгоритма	50
Количество итераций генетического алгоритма	100
Количество состояний N_t	10
Нижняя граница функции правдоподобия \underline{G}	-250
Верхняя граница функции правдоподобия \bar{G}	-90
R_0 , млн. руб. (для k -ого запуска рассматривались k первых компонент,	$(1200, 1800, 1200, 1000, 500, 7000)^T$

$k = 1, \dots, 6$	
-------------------	--

Табл. 5. Параметры запусков генетического алгоритма и метода динамического программирования с разным значением \underline{G} .

Параметр	Значение параметра
Количество заемщиков (k)	6
Количество внешних параметров (N)	4
Временной период (T)	12 кварталов (3 года)
Вектор кредитов S , млн. руб.	$(15, 18, 21, 25, 34, 100)^T$
Численность популяции генетического алгоритма	50
Количество итераций генетического алгоритма	100
Количество состояний N_t	10
Нижняя граница функции правдоподобия \underline{G} (фиксированная для каждого запуска)	[-270;-250;-230;-210;-190;-170;-150;-130;-110;90]
Верхняя граница функции правдоподобия \bar{G}	-70
R_0 , млн. руб.	$(1200, 1800, 1200, 1000, 500, 7000)^T$

График, изображенный на Рис. 8, иллюстрирует зависимость скорости работы методов от объема рассматриваемого портфеля. В данном случае метод динамического программирования (изображен сплошной линией) демонстрирует большую производительность и значительно меньшую зависимость скорости работы от количества рассматриваемых заемщиков, чем генетический алгоритм (изображен пунктирной линией). Графики, представленные на Рис. 9, демонстрируют способность методов строить сценарии, обеспечивающие наибольший уровень финансовых потерь. В некоторых случаях при различных фиксируемых значениях \underline{G} генетический алгоритм (изображен пунктирной линией) находил лишь локальные минимумы задачи обратного стресс-тестирования, при этом метод динамического программирования (сплошная линия) сходил к глобальному решению гораздо чаще.

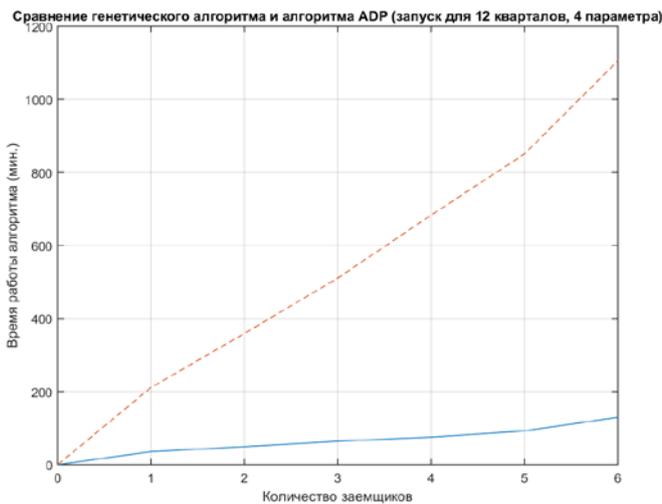


Рис. 8. Скорость работы генетического алгоритма (пунктирная линия) и приближенного динамического программирования (сплошная линия).

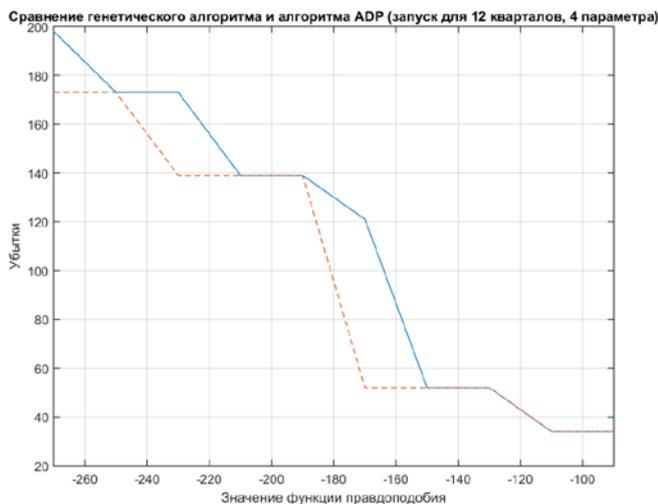


Рис. 9. Сходимость к оптимальным макроэкономическим сценариям генетического алгоритма (пунктирная линия) и метода приближенного динамического программирования (сплошная линия).

VI. ЗАКЛЮЧЕНИЕ

В статье приведена постановка задачи обратного стресс-тестирования и представлен алгоритм, решающий данную задачу для кредитного портфеля, состояние которого определяется системно-динамическими моделями заемщиков. Алгоритм основан на методе приближенного динамического программирования, каждый шаг которого сводится к решению гладких задач условной оптимизации при помощи метода последовательного квадратичного программирования. Сформулированные в работе утверждения обосновывают корректность разработанного метода и дают представление о его применимости к негладким задачам оптимизации, возникающим в обратном стресс-тестировании. Реализация описанного алгоритма осуществлялась с помощью Matlab Simulink.

Проведенный анализ численных результатов работы рассматриваемого метода позволяет сделать вывод о его эффективности в реализации процедуры обратного стресс-тестирования и наличии преимуществ перед генетическими алгоритмами с точки зрения скорости работы и способности построения сценариев, приводящих к наибольшим финансовым потерям. Дальнейшие исследования могут быть направлены на улучшение разработанного метода и более детальное изучение его свойств.

БИБЛИОГРАФИЯ

1. Bollerslev, T. Generalized autoregressive conditional heteroskedasticity // *Journal of Econometrics*. - 1986. - N 31. - P. 309-328.
2. Engle, R. F. Autoregressive conditional heteroskedasticity with estimates of the UK inflation // *Econometrica*. - 1982. - N 50. - P. 987-1008.
3. Flood M. D., Korenko G. G. Systematic scenario selection: stress testing and the nature of uncertainty // *Office of financial research*. 2013.
4. Forrester J. W. *Industrial Dynamics* / MIT Press. 1961.
5. Glasserman P., Kang C., Kang W. Stress scenario selection by empirical likelihood // *Office of financial research*. 2012.
6. Guidelines on stress testing / *Committee of European banking supervisors*. 2010.
7. Gurný P., Gurný M. Comparison of credit scoring models on probability of default estimation for us banks // *Prague economic papers*. 2013.
8. MATLAB. *The Language of Technical Computing*. Using MATLAB. MathWorks Inc., 1999.
9. MATLAB. URL.: <https://www.mathworks.com/>.
10. Powell W.B. *Approximate dynamic programming: Solving the curses of dimensionality* / Wiley. 2007.
11. Principles for sound stress testing practices and supervision / *Basel committee on banking supervision*. 2009.
12. Schwarz, G. Estimating the dimension of a model // *Annals of Statistics*. - 1978. - N 2. - P. 461-464.
13. Schwefel, H. P. *Numerical Optimization of Computer Models* / Wiley. 1981.
14. Sterman J. D. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Boston: McGraw-Hill Companies. 2000.
15. Bellman R. *Dynamic programming*. - M.: Foreign Languages Publishing House, 1960.
16. Boks Dzh., Dzhenkins T. : *Time Series Analysis, Forecasting and Control*, M.: Mir Publishers, 1974.
17. Izmailov A. F. Solodov M. V. *Numerical Methods of Optimization*. - M.: Fizmatlit, 2008.
18. Kurennoy D. S., Golembiovskiy D. Yu. Estimating the probability of oil company default based on system dynamics model // *Issues of Risk Analysis*. 2018. №02. P. 86-92.
19. Kurennoy D. S., Golembiovskiy D. Yu. Construction of business default scenarios based on system dynamics models // *Journal of Applied Informatics*. 2017. №05. P. 11-28.
20. Kurennoy D. S., Golembiovskiy D. Yu. System dynamics credit risk model of an oil company // *Issues of Risk Analysis*. 2017. №01. P. 6-22.
21. Tot'myanina K. M. Review of models of default probability // *Financial risk management journal*. - 2011. - №01 (25). - P. 12-24.
22. Ukazanie Banka Rossii ot 07.12.2015 N 3883-U. URL: http://www.consultant.ru/document/cons_doc_LAW_190733/
23. Ukazanie Banka Rossii ot 15.04.2015 N 3624-U. URL: http://www.consultant.ru/document/cons_doc_LAW_180268/.
24. Cisar I. F. *MATLAB Simulink. Computer modeling of economy*. - M.: SOLON-PRESS, 2008. - 256 pp.

Сведения об авторе:

Куренной Дмитрий Святославович: аспирант кафедры исследования операций факультета вычислительной математики и кибернетики МГУ им. М.В. Ломоносова.
E-mail: d.kurennoy@yandex.ru

Algorithm for solving the reverse stress testing problem of the bank's loan portfolio on basis of system dynamic models of borrowers

D.S. Kurennoy

Abstract—The key mechanisms to determine the effectiveness of bank's activities is the credit risk management. One of the main procedures recommended to banks for risk assessment is reverse stress testing. Reverse stress testing is the construction of realistic scenarios that lead to a given level of financial loss, or the formation of realistic scenarios that maximize bank losses. Knowledge of such scenarios allows banks to mitigate the consequences of their implementation.

The current credit risk assessment models are not suitable for solving the problems of reverse stress testing. Widely known models do not take into account the structure of specific companies, do not allow investigating the development of crisis scenarios in time and assuming a large sample of data on similar enterprises. To avoid of the noted disadvantages of the traditional approaches allows the use system-dynamic models of borrowers. System dynamics provides the possibility of reproducing the enterprise's structure under study in the form of continuously interacting elements and external factors. The links between the elements are described by functional dependencies and differential equations that determine the dynamics of the company and the degree of its stability in relation to various macroeconomic scenarios. The article is devoted to approximate dynamic programming algorithm, which solves the reverse stress testing problem for the bank's loan portfolio on the basis of system-dynamic models of borrowers. The algorithm uses classical quasi-Newton optimization methods. The results of the method work are compared with the results of genetic algorithms. The statements proved in the article justify correctness of the algorithm and give an idea of its applicability to nonsmooth optimization problems arising in the context of reverse stress testing and system dynamics. The main tool for implementing the described algorithm is Matlab.

Keywords — system dynamics, credit risk, reverse stress testing, approximate dynamic programming, simulation modelling, quasi-Newton optimization methods, Matlab, Simulink.