

Том II. Развитие модели QSG: версии v6.x и далее

Настоящий текст продолжает изложение, начатое в документе *name3.tex*, где фиксиированы онтология спектральной среды, геометрическая таблица, версия QSG v5.0 и введён минимальный Spectral Lab v1. Здесь собираются рабочие записки и расширения, относящиеся к будущим версиям модели (условно QSG v6.x и далее), включая углублённый Spectral Lab, генераторы с циклами, d-блок, ядерный мост и дальнейшее развитие спектральной гравитации.

0.1. Обзор направления QSG v6.x

В качестве отправной точки принимается следующая конфигурация:

- **Зелёная зона QSG v5.0:** геометрический атом H–Kr, D/A-индексы и роли, деревьевой Grower, функционал сложности (v1 и FDM) с константами α_{tree} , а также мягкость $s(Z)$ для ростовой динамики.
- **Spectral Lab v1 (QSG v6.0 в зачатке):** одномерный оператор \hat{H} на решётке, спектр, DOS/LDOS и игрушечный функционал F_{levels} с FDM-прокси.
- **Жёлтая зона:** континуальный оператор $\hat{H}[\theta]$, честный DOS/LDOS, crossing-слой сложности, полноценный d-блок, строгий мост $\text{geom} \leftrightarrow F_{\text{nuc}}$.
- **Красная зона:** спектральная гравитация и космология, многоуровневая солитонная онтология и полная интеграция всех слоёв в единую спектральную теорию.

0.2. HETERO-1A как прикладной контур QSG

Линия HETERO-1A (acid / hetero growth) выросла из разделов *name3* о геометрических функционалах и служит прикладной проверкой идей QSG. Здесь мы переносим абстрактные постулаты в конкретные скрипты (*analysis/chem/**) и следим, чтобы экспериментальные метрики оставались совместимыми с F_{levels} и F_{geom} :

- **Связь со спектральной гравитацией.** Пары классов (alcohol/ether, первичный/вторичный амин и т.д.) трактуются как локальные вариации спектра Среды. Любая метрика (ROC-AUC, collision rates) описывает, насколько две конфигурации различимы в духе *name3*.
- **Tie-aware AUC.** Введён взвешенный Mann–Whitney AUC с агрегированием по блокам равных энергий. Это реализует требование *name3* об инвариантности к перестановкам внутри спектрально неразличимых состояний и устраняет зависимость от порядка.
- **Динамический neg-control gate.** Вместо фиксированной оценки 0.60 мы берём точную комбинаторную квантиль $q_{0.95}^{\text{null}}(m, n)$ и добавляем регулируемый запас (по умолчанию 0.06). Такой гейт интерпретируется как “спектральный запрет” и логируется (`fp_neg_auc_gate`, `fp_neg_auc_null_q`, `fp_neg_auc_slack`).
- **Slack и Grower/FDM.** Колонка `fp_neg_auc_slack` показывает, насколько далеко мы от порога. Если $\text{slack} \approx 0$, это сигнал к пересмотру параметров grower/FDM или увеличению margin, чтобы не нарушить спектральные ограничители *name3*.

Таким образом, HETERO-1A выступает “витриной” томов *name3/name4*: мы валидируем спектральные постулаты на конкретных гетеро-семействах, логируем пороги/квантили и используем ту же терминологию (инварианты, функционалы, slack), что и в теоретической части.

Задача тома II — пошагово продвигать модель от текущего состояния к более полному спектральному описанию, при этом каждый шаг должен быть *привязан к коду и численным экспериментам*, а не оставаться на уровне чисто словесных деклараций.

0.3. Spectral Lab v1 как R&D-площадка

Цель Spectral Lab v1 — служить минимальной лабораторией для проверки идей о спектральном операторе \hat{H} и функционале уровней F_{levels} , а также для отработки связи между спектральной картиной и FDM-приближениями.

В *name3.tex* описан базовый одномерный оператор

$$H = -\frac{1}{2m} \frac{d^2}{dx^2} + V(x)$$

на равномерной решётке, реализованный в модуле `core/spectral_lab_1d.py`, и две версии функционала уровней: спектральную $F_{\text{levels}}^{(\text{spec})}$ и FDM-прокси $F_{\text{levels}}^{(\text{FDM})}$ (модуль `core/f_levels_1d.py`). Здесь мы фиксируем план экспериментов, которые должны сделать Spectral Lab содержательным шагом в сторону настоящего $\hat{H}[\theta]$.

0.3.1. Эксперимент SL-1: разрешение решётки и согласие spec/FDM

Первая задача Spectral Lab v1 — количественно проверить, насколько FDM-прокси для F_{levels} приближают спектральные значения при разумных параметрах. Для этого вводится эксперимент SL-1.

В реализации QSG v6.0 эксперимент SL-1 оформлен скриптом `analysis/scan_spectral_lab_1d.py`, который для гармонического потенциала на отрезке $[-5, 5]$ при разных N сравнивает $F_{\text{levels}}^{(\text{spec})}$ и $F_{\text{levels}}^{(\text{FDM})}$ и записывает численные значения и относительную ошибку в файлы `results/spectral_lab_1d.csv` и `results/spectral_lab_1d_resolution.csv`.

Постановка.

- Выбирается один фиксированный потенциал (например, гармонический осциллятор) с разумными параметрами m, k .
- Рассматривается набор размеров решётки $N \in \{100, 200, 400, 800, \dots\}$ при фиксированном интервале по x .
- Для каждой решётки строится оператор H , находится спектр и вычисляется $F_{\text{levels}}^{(\text{spec})}$ с выбранной весовой функцией $w(E)$.
- По тем же данным считается FDM-прокси $F_{\text{levels}}^{(\text{FDM})}$ и относительная ошибка

$$\varepsilon(N) = \frac{|F_{\text{levels}}^{(\text{spec})} - F_{\text{levels}}^{(\text{FDM})}|}{|F_{\text{levels}}^{(\text{spec})}|}.$$

Ожидаемый результат. Ожидается, что при увеличении N ошибка $\varepsilon(N)$ будет падать или по крайней мере стабилизироваться на приемлемом уровне для разумных значений N . Этот эксперимент задаёт *оперативный диапазон* размерностей решётки, в котором FDM-прокси может считаться достаточно точным для R&D-целей.

Фактический исход (Spectral Lab v1). Численный запуск эксперимента SL-1 для гармонического осциллятора показал, что спектральный функционал уровней $F_{\text{levels}}^{(\text{spec})}$ стабилизируется около значения порядка 1.25 при увеличении числа узлов решётки N , тогда как FDM-прокси $F_{\text{levels}}^{(\text{FDM})}$ остаётся на уровне порядка 0.30. Относительная ошибка $\varepsilon(N) \approx 0.75$ практически не меняется для $N \in \{100, 200, 400, 800\}$. Это означает, что текущая форма FDM-функционала для F_{levels} в Spectral Lab v1 даёт существенно смещённую оценку и требует пересмотра.

SL-1b: линейная калибровка FDM-прокси. Для уменьшения систематического смещения между $F_{\text{levels}}^{(\text{spec})}$ и $F_{\text{levels}}^{(\text{FDM})}$ в Spectral Lab v1 введена линейная FDM-модель

$$F_{\text{levels}}^{(\text{FDM,lin})} = a F_{\text{naive}} + b,$$

где F_{naive} — прежняя FDM-аппроксимация по потенциалу $V(x)$. Калибровочный скрипт `analysis/calibrate_f_levels_fdm_1d.py` подбирает коэффициенты (a, b) методом наименьших квадратов по данным SL-1 (гармонический потенциал, несколько значений N). В текущих расчётах получается почти постоянная поправка: a мало по модулю, $b \approx 1.2557$, а среднеквадратичная ошибка относительно спектрального F_{levels} падает до $\sim 6 \cdot 10^{-4}$ на калибровочном наборе. Это подтверждает, что исходный F_{naive} содержит мало информативной структуры и требует более содержательного переопределения.

0.3.2. Эксперимент SL-2: игрушечная спектральная “жёсткость” $\chi_{\text{spec}}^{(1D)}$

Второй эксперимент направлен на поиск одномерного аналога спектральной “жёсткости” или электроотрицательности χ_{spec} . Идея состоит в том, чтобы изучить семейство потенциалов $V(x; \lambda)$ с параметром жёсткости λ и посмотреть, можно ли определить функционал $\chi_{\text{spec}}^{(1D)}(\lambda)$, монотонно отражающий изменения жёсткости.

Постановка.

- Выбирается параметризованное семейство потенциалов $V(x; \lambda)$ (например, гармонический осциллятор с разными k или двойная яма с меняющейся высотой/шириной барьера).
- Для каждого значения λ строится спектр оператора $H(\lambda)$ и вычисляется $F_{\text{levels}}^{(\text{spec})}(\lambda)$ с фиксированной весовой функцией $w(E)$.
- Определяется игрушечный показатель

$$\chi_{\text{spec}}^{(1D)}(\lambda) = \frac{F_{\text{levels}}^{(\text{spec})}(\lambda)}{\mathcal{N}(\lambda)},$$

где $\mathcal{N}(\lambda)$ — выбранная нормировка (например, число уровней в определённом энергетическом окне или простая функция параметров потенциала).

- Анализируется монотонность и стабильность $\chi_{\text{spec}}^{(1D)}(\lambda)$ при изменении λ .

Ожидаемый результат. Если удаётся подобрать такую нормировку $\mathcal{N}(\lambda)$ и вес $w(E)$, при которых $\chi_{\text{spec}}^{(1D)}(\lambda)$ ведёт себя монотонно и устойчиво при изменении жёсткости потенциала, это даёт численный прототип для будущих определений спектральной электротрицательности в более реалистичных конфигурациях.

Выбор прототипов $\chi_{\text{spec}}^{(1D)}$ и спектральной мягкости. Численные результаты для семейства гармонических потенциалов $V(x; \lambda) = 12\lambda x^2$ показывают, что величины $\chi_0(\lambda)$, $\chi_{\text{avg},5}(\lambda)$, $\chi_{\text{avg},10}(\lambda)$ и энергетический хвост $\chi_{\text{tail}}(\lambda)$ монотонно растут при увеличении λ , тогда как спектральный функционал уровней $F_{\text{levels}}^{(\text{spec})}(\lambda)$ с фиксированной гауссовой весовой функцией убывает.

В качестве игрушечного прототипа спектральной “жёсткости” в одномерной модели мы фиксируем

$$\chi_{\text{spec}}^{(1D)}(\lambda) \equiv \chi_{\text{avg},10}(\lambda) = \frac{1}{10} \sum_{k=0}^9 E_k(\lambda),$$

то есть среднюю энергию первых десяти уровней гамильтониана $H(\lambda)$. Эта величина монотонно растёт с λ и ведёт себя как $\chi_{\text{spec}}^{(1D)} \sim \sqrt{\lambda}$, что согласуется с ожидаемой зависимостью спектра гармонического осциллятора от параметра жёсткости.

Одновременно функционал

$$s_{\text{spec}}^{(1D)}(\lambda) \equiv F_{\text{levels}}^{(\text{spec})}(\lambda; w(E)) = e^{-E^2/2},$$

который убывает при увеличении λ , естественно интерпретировать как спектральную “мягкость” одномерного потенциала. Таким образом, в Spectral Lab v1 появляется пара со-пряжённых характеристик: спектральная жёсткость $\chi_{\text{spec}}^{(1D)}$ и спектральная мягкость $s_{\text{spec}}^{(1D)}$, которые качественно отражают поведение спектра при изменении параметра жёсткости λ .

0.4. Дальнейшая дорожная карта тома II

В следующих разделах предполагается детализировать и реализовать:

- расширение Spectral Lab до двумерных операторов и простых “атомоподобных” потенциалов;
- постепенный отход d-блока от чисто эмпирической калибровки по Паулингу к собственным геометрическим и спектральным моделям;
- генератор молекулярных графов с циклами и интеграцию crossing-слоя сложности;
- уточнение и углубление ядерного модуля и спектрального моста между геометрической таблицей и F_{nuc} .
- равновесный химический контур для алканов (tree-only): fixed- N МСМС, диагностика mixing, строгая coverage и продуктовый pipeline C15/C16.

0.5. Пакет CHEM-VALIDATION: равновесные алканы (C4–C16)

В рамках химического контура QSG реализован и закрыт (как продуктовый стенд) контур валидации на классе *tree-only* алканов: связные деревья на N вершинах (углероды), $c = N - 1$ рёбер, цикломатическое число 0, ограничение валентности $\max \deg \leq 4$. Цель — получать воспроизводимую равновесную меру по топологиям при фиксированном N и использовать её как эталон для ранжирования изомеров по “энергии сложности” (Mode A: FDM-only на деревьях).

Топологии, пометки и дегенерация. Рассматриваются *помеченные* деревья (labeled, вершины $1..N$) и индуцированные ими *непомеченные* топологии (unlabeled). При переходе к топологиям учитывается комбинаторная дегенерация

$$g(\tau) = \frac{N!}{|\text{Aut}(\tau)|},$$

где $\text{Aut}(\tau)$ — группа автоморфизмов топологии τ . Это обязательный множитель при со-поставлении “энергий топологии” с частотами.

Рост ≠ равновесие. Частоты, наблюдаемые в grower-процессе, отражают прежде всего *proposal/kernel bias* и не являются равновесием при фиксированном N . Равновесная мера P_{eq} строится отдельным слоем: fixed- N MCMC по пространству помеченных деревьев (leaf-rewire + поправка Гастингса). Для малых N используется exact-база (Prüfer), что позволяет проверять корректность MCMC “по истине”.

Инвариантность энергии на деревьях. Критический инженерный инвариант: энергия (FDM) на tree-only должна быть *перестановочно-инвариантной*. Для этого перед вычислением FDM выполняется канонизация дерева (детерминированное переименование вершин), после чего для одной и той же топологии дисперсия энергии по различным пометкам стремится к нулю. Этот шаг восстановил согласованность P_{pred} и exact-базы на малых N и устранил ложные расхождения, вызванные label-dependence.

DoD-метрики mixing и coverage. Для больших N (C15/C16) вместо “стрельбы вслепую” бюджет шагов калибруется по guardrail-метрикам (несколько стартов, несколько цепей):

- $KL_{\text{max_pairwise}} \leq 0.01$ (макс. KL между цепями),
- $KL_{\text{split_max}} \leq 0.01$ (KL между первой/второй половинами),
- $\hat{R}_{\text{energy}} \leq 1.05$,
- $ESS_{\text{energy}} \geq 500$,
- **строгая coverage:** n_{unique} совпадает с ожидаемым числом непомеченных топологий (C15: 4347; C16: 10359).

В CHEM-VALIDATION-5 бюджет задаётся как *total steps* по всем $\#starts \times \#chains$; число шагов на цепь: $\text{steps_per_chain} = \text{steps_total}/(\#starts \cdot \#chains)$. Для финальных прогонов фиксируются также `thin=10` и `burnin_frac=0.30`.

Distributed-контур (EQ-DIST-1). Для устойчивости к тайм-лимитам и масштабирования на произвольное железо введены *work units* (аналог GIMPS): генератор задач (`make_tasks`), воркер (`worker`) и агрегатор (`aggregate`) с режимом `--status` (READY/MISSING) и опциональной двойной верификацией на выбранных шагах. Энергия переиспользуется через персистентный `energy-cache`, а агрегаты фиксируются `commit+push`.

Финальный результат C15/C16 (Mode A, fixed budget). Контур CHEM-VALIDATION-5 закрыт на строгих DoD-порогах и 100% coverage:

Метрика	C15 (финал)	C16 (финал)
steps_total	1.60×10^8	3.20×10^8
steps_per_chain	1.6×10^7	3.2×10^7
$KL_{\max_pairwise}$	0.004114	0.004858
$KL_{\text{split_max}}$	0.008336	0.009774
$\hat{R}_{\text{energy}} (\max)$	1.000000	1.000001
$ESS_{\text{energy}} (\min)$	3.79×10^5	6.73×10^5
Coverage (n_{unique})	4347/4347	10359/10359

Комментарий о “статистическом полу” KL_{split} . На больших пространствах (C16: $K \approx 10^4$ топологий) KL_{split} приближается к нижнему уровню, обусловленному конечной длиной цепей и дискретностью эмпирических частот. Практически это означает: дальнейшее увеличение бюджета даёт уменьшение $KL_{\max_pairwise}$ быстрее, чем KL_{split} . Инженерный порог 0.01 остаётся разумным gate, но требует интерпретации вместе с coverage и \hat{R}/ESS .

Следующий инженерный шаг. Далее контур упаковывается в стабильный продуктивный pipeline: единые CLI/конфиги, детерминированные seeds, единый формат отчётов, регрессионные тесты на малых N (exact vs MCMC) и расширение за пределы tree-only (циклы/конформации) с сохранением раздельной трактовки proposal bias и равновесия.

0.6. Geom–nuclear–FDM: первый связанный стенд

На уровне версий QSG v5.x/v6.0 собран комбинированный слой geom–nuclear–complexity: таблица `geom_nuclear_complexity_summary.csv`, объединяющая геометрический атом (роли, χ_{spec} , индексы D/A), FDM-комплексность (средние по росту деревьев) и ядерные индикаторы. Комбинированная таблица формируется скриптом `analysis/analyze_geom_nuclear_c`, который объединяет геометрические индексы, FDM-комплексность и ядерные таблицы (`geom_isotope_bands.csv`, `geom_nuclear_map.csv`), а также вычисляет производные поля ('band_width', N_{best}/Z , нейтронный избыток). На этом слое запущен первый анализ корреляций между нормированной FDM-комплексностью и ядерными характеристиками (`analysis/analyze_fdm_vs_nucleus.py`, отчёт `results/fdm_vs_nucleus_stats.txt`).

Для элементов H–He глобальные Spearman-корреляции показывают умеренные связи между нормированной FDM-сложностью и ядерными параметрами (шириной полосы 'band_width', отношением 'Nbest_over_Z' и нейтронным избытком), а по ролям и отдельным группам (terminator/bridge/hub, d-блок, living hubs) картина остаётся шумной из-за малых выборок. Этот слой зафиксирован как R&D-стенд: он демонстрирует наличие слабых/умеренных связей между геометрической сложностью и ядерными характеристиками, но для строгих утверждений требуется более полный и однородный ядерный датасет.

0.7. Пакет CY-1: первый стенд циклов и crossing-proxy (QSG v6.x)

В рамках пакета CY-1 для QSG v6.x введён первый стенд циклов: отдельный скрипт `analysis/scan_cycles_vs_params.py` исследует область параметров роста, в которой стохастический Grower начинает порождать графы с ненулевым цикломатическим числом. На этом уровне вводится простой crossing-proxy (циклонагрузка на вершину), который служит тренировочной площадкой для будущего топологического слоя сложности (crossing-number и родственные инварианты).

Режимы loopy-growth (CY-1-A/B). После подтверждения деревьевого предела QSG v5.0 в пакете CY-1 введён R&D-режим роста с циклами (loopy-growth). На уровне ядра расширены параметры роста `GrowthParams` флагом `allow_cycles` и двумя числовыми параметрами (`max_extra_bonds`, `p_extra_bond`), которые по умолчанию выключены. Дополнительный слой *loopy overlay* добавляет случайные связи между уже существующими вершинами поверх готового дерева и тем самым порождает графы с ненулевым цикломатическим числом.

На основе скана параметров роста (`results/cycle_param_scan.csv`) выбраны эталонные режимы CY-1-А (умеренная доля циклов) и CY-1-В (агрессивный циклонасыщенный рост). Скрипт `analysis/analyze_loopy_modes.py` измеряет для них долю графов с циклами, распределение цикломатического числа и среднюю циклонагрузку на вершину, что служит первым численным прототипом crossing-слоя топологической сложности. Слой FDM-сложности при этом расширен R&D-вариантом `fdm_loopy`: для заданной FDM-комплексности $C_{\text{FDM}}(G)$ вводится штраф

$$C_{\text{FDM}}^{(\text{loopy})}(G) = C_{\text{FDM}}(G) \left(1 + \alpha_{\text{cycle}} \mu(G) + \alpha_{\text{load}} \mu(G) n(G) \right),$$

где $\mu(G)$ — цикломатическое число, $n(G)$ — число вершин. Для деревьев ($\mu(G) = 0$) закон совпадает с базовым FDM-режимом, а для графов CY-1-А/В дополнительный множитель делает циклы «дорогими» с точки зрения FDM-закона сложности. Дополнительно для малых графов (обычно $n \leq 8$) вводится игрушечный crossing-number $cr_{\text{circle}}(G)$ в модели “вершины на окружности, рёбра — хорды”, реализованный в модуле `core/crossing.py`. Анализ по режимам CY-1-А/В показывает, что плотность пересечений $cr_{\text{circle}}(G)$ разумно коррелирует как с циклонагрузкой $\mu(G)/n(G)$, так и с фактором штрафа в FDM-законе, что служит первым численным подтверждением согласованности crossing-aware FDM-слоя с дискретным прототипом пересечений. Для дальнейшего R&D введён расширенный backend `fdm_loopy_cross`, в котором поверх циклового штрафа используется дополнительный множитель $(1 + \beta_{\text{cross}} \cdot \text{crossing_proxy}(G))$, где при малых размерах графа proxy строится по $cr_{\text{circle}}(G)$, а при больших используется циклонагрузка. Параметр β_{cross} калибруется отдельно и не влияет на базовый FDM-режим v5.0.

Пакет TEMP-1: температура среды и шум роста. В ростовых параметрах QSG v6.x введён явный параметр температуры T , который деформирует эффективную вероятность продолжения ветви p_{continue} так, что при $T = 1$ воспроизводится базовый деревьевый режим QSG v5.0, при $T < 1$ деревья и loopy-графы в среднем растут длиннее, а при $T > 1$ — короче и легче распадаются. Конфигурации среды `growth_env_cold.yaml` и `growth_env_hot.yaml` задают первые “холодные” и “горячие” режимы для loopy-роста, а стенд `analysis/scan_temperature` количественно описывает зависимость среднего размера и циклонагрузки от T для элементов C, Si, O, S. Этот слой интерпретируется как первый технический шаг к модели манаховского “шума среды” и спектральной мягкости/жёсткости связей.