

Validierungsbericht — Erkennung der Restriktionsstellen

Werk: codex-vivum · **Version:** v1.0 · **Datum:** 2026-05-12 **Autor:** Thomas Schüller · ORCID 0009-0003-9799-6747 · CC BY 4.0

Dieser Bericht setzt Abschnitt 4.2 der im Papier vor-registrierten Validierung um. Er dokumentiert zugleich einen dabei gefundenen und behobenen Fehler.

Anlage

Die Schnittstellen-Erkennung des Werkzeugs wurde nach Python portiert und gegen `Bio.Restriction` aus Biopython verglichen — eine unabhängige Implementierung, deren Enzymdaten aus REBASE stammen.

Testmenge. 19 Sequenzen (davon zwei echte Klonierungsvektoren): zwölf Zufallssequenzen über die Längen 1, 3, 5 und 10 kb bei GC-Gehalten von 35, 50 und 65 Prozent, dazu fünf konstruierte Randfälle (Erkennungsstelle am Sequenzanfang, am Ende, über die Naht zirkulärer DNA, dicht aufeinanderfolgende Stellen, mehrdeutige Muster). Deterministisch über festen Zufallsstartwert (4273).

Umfang. 43 Enzyme, alle in `Bio.Restriction` vorhanden. Je Enzym und Sequenz ein Vergleich für lineare und einer für zirkuläre DNA: 1.462 Positionsvergleiche, dazu 166 Vergleiche der Fragmentgrößen.

Befund — überlappende Erkennungsstellen wurden übersehen

Der erste Durchlauf ergab 99,18 Prozent Übereinstimmung. Alle Abweichungen betrafen ein einziges Enzym: **NlaIV** mit der Erkennungssequenz GGNCC. Das Werkzeug fand durchgängig *weniger* Stellen als die Referenz.

Die Prüfung der 28 zusätzlich gefundenen Stellen zeigte, dass es sich um gültige GGNCC-Treffer handelt. Beispiel aus einer Testsequenz: An der Stelle GGGTCCC überlappen zwei Treffer — GGGTCC ab Position 76 und GGTCCC ab Position 77. Der reguläre Ausdruck des Werkzeugs setzte nach einem Treffer hinter dessen Ende fort und übersprang den zweiten.

Das war ein echter Fehler. Eine reale Schnittstelle wurde nicht angezeigt, und die daraus berechneten Fragmente waren entsprechend falsch.

Reichweite. Ein systematischer Test aller 43 Enzyme ergab, dass sich sieben Muster selbst überlappen können: ClaI (ATCGAT), NheI (GCTAGC), NotI (GCGGCCGC), NsiI (ATGCAT), PvuI (CGATCG), SphI (GCATGC) und NlaIV (GGNCC). Bei den sechs Enzymen ohne Mehrdeutigkeit ist die Überlappung in Zufallssequenzen so selten, dass sie in der Testmenge nicht auftrat — der Fehler war dort latent vorhanden und wäre an realen Sequenzen aufgetreten.

Behebung. Der Suchausdruck wird nun in einen Lookahead gefasst, der nichts verbraucht; der Suchlauf rückt danach um eine Position vor statt um die Trefferlänge. Damit werden überlappende Stellen gefunden.

Zwei weitere Befunde

Nach der Behebung des Überlappungsfehlers ergab die Fortführung der Prüfung zwei weitere Mängel, die beide die Aussagekraft der Prüfung selbst betrafen.

Die Enzymliste war doppelt geführt

Die erste Fassung dieses Skripts enthielt eine von Hand übertragene Zweitfassung der Enzymtabelle. Der Abgleich mit der Quelle zeigte neun Abweichungen: Geprüft wurden Enzyme, die es im Werkzeug nicht gibt (NlaIV, ApoI, BstYI, NsiI, SacII, SspI, MspI), während neun tatsächlich vorhandene Enzyme (HpaII, MboI, MfeI, PacI, PciI, SbfI, BsaI und weitere) nie erfasst wurden. Der zuvor berichtete Wert bezog sich damit teilweise auf eine Bibliothek, die so nicht existiert.

Behoben, indem das Skript die Tabelle nun aus `src/lib/restriction.js` liest. Weicht die Datei ab oder fehlt sie, bricht der Lauf ab. Eine zweite Fassung derselben Daten wird nicht mehr geführt.

Typ-IIS-Enzyme schnitten im Gegenstrang falsch

Der Lauf über die tatsächliche Bibliothek ergab 99,04 Prozent. Alle Abweichungen betrafen BsaI, und zwar nicht in der Anzahl, sondern in der Position: Jede Stelle im Gegenstrang lag um genau 4 Nukleotide daneben.

BsaI ist ein Typ-IIS-Enzym und schneidet außerhalb seiner Erkennungssequenz, nach REBASE GGTCTC(1/5) — ein Nukleotid hinter der Stelle im oberen, fünf im unteren Strang. Das Datenmodell des Werkzeugs kannte nur *einen* Schnittversatz. Für eine Erkennungsstelle im Gegenstrang wird aber der zweite gebraucht. Die Differenz von 4 ist genau der Abstand zwischen beiden Versätzen.

Behoben durch ein zusätzliches Feld `cutBottom` im Enzymeintrag und eine gesonderte Rechnung für Typ-IIS-Enzyme im Gegenstrang. Eine Prüfung im Testsatz stellt sicher, dass kein künftiges Typ-IIS-Enzym ohne diesen zweiten Wert eingetragen wird.

Bewertung dieser beiden Befunde

Der erste betraf die Prüfung, nicht das Werkzeug — aber er hätte beinahe dazu geführt, dass ein falsches Ergebnis als Bestätigung berichtet worden wäre. Der zweite betraf das Werkzeug und wäre ohne die Korrektur des ersten nie sichtbar geworden, weil BsaI in der abgetippten Liste fehlte.

Ergebnis nach der Korrektur

Prüfung	Vergleiche Übereinstimmung	
Schnittpositionen, lineare DNA	817	817 (100,00 %)
Schnittpositionen, zirkuläre DNA	817	817 (100,00 %)
Mehrdeutige Erkennungssequenzen	3 Enzyme	3 (100 %)
Fragmentgrößen	178	178 (100,00 %)

Die Fragmentlängen ergeben in allen Fällen die Sequenzlänge.

Die mehrdeutigen Enzyme der Bibliothek im Einzelnen: Aval (CYCGRG), BstEII (GGTNACC), HinfI (GANTC).

Abgleich mit publizierten Angaben

Der Testmenge wurden die Klonierungsvektoren **pUC19** (2.686 bp, GenBank L09137) und **pBR322** (4.361 bp, J01749.1) hinzugefügt. Ihre Herkunft und die Prüfung vor der Verwendung sind in `sequenzen/HERKUNFT.md` dokumentiert.

Damit ergibt sich ein Abgleich gegen eine **dritte** Quelle: Für pUC19 ist publiziert, dass die zehn Polylinker-Enzyme EcoRI, SacI, KpnI, SmaI, BamHI, XbaI, SalI, PstI, SphI und HindIII je genau einmal schneiden. Das Werkzeug findet für alle zehn genau eine Schnittstelle. Diese Prüfung hängt nicht an der Referenzimplementierung, sondern an der veröffentlichten Karte des Vektors.

Gesamtumfang nach Aufnahme der Vektoren: **1.634 Positionsvergleiche**, alle identisch.

Was dieser Bericht nicht zeigt

- Verglichen wurde gegen eine unabhängige Implementierung, deren Enzymdaten aus derselben Quelle (REBASE) stammen. Übereinstimmung belegt korrekte Umsetzung der Daten, nicht deren Richtigkeit.
- Geprüft wurden Positionen und Fragmentlängen. Die Darstellung im simulierten Agarosegel und die Angaben zu Überhängen sind nicht Gegenstand dieser Prüfung.
- Der Abgleich mit NEBcutter selbst steht weiterhin aus; geprüft wurde gegen Bio.Restriction und gegen die publizierte Polylinker-Karte von pUC19.
- Methylierungsabhängigkeit (etwa bei DpnI) wird vom Werkzeug nicht modelliert und war nicht Gegenstand des Vergleichs.

Bewertung

Der Wert dieser Prüfung liegt weniger in der Bestätigung als im Fund. Ohne den Vergleich gegen eine unabhängige Implementierung wäre der Fehler bei NlaIV nicht aufgefallen — er äußert sich nicht als Absturz, sondern als stilles Fehlen einzelner Schnittstellen.

Reproduktion

```
pip install -r requirements.txt
python3 restriktion_validation.py
```

Das Skript endet mit Rückgabewert 0 nur bei vollständiger Übereinstimmung. Vollständige Ausgabe: `restriktion_output.txt`.