

Å fryse et laboratorium: korpus, fasit og baseline for skjærgårdsruting — og kartet som endret seg underveis

Ola Jacob Mjøen Iversen

Fellag AS (BoatNav), Kongsberg, Norge · olajacob@fellag.as

BoatNav Research Note 2026-03 · versjon 1.3, 1. august 2026 · preprint — ikke fagfelle­vurdert

Abstract. We report the construction and independent verification of a frozen experimental apparatus for maritime route planning: a 273-instance benchmark corpus, a reference solution with a *measured* rather than assumed error budget, and a characterization of the incumbent planner against both. Every artefact carries a cryptographic identity and a gate that refuses execution on a mismatch. The incumbent returns *no route* for **61 of 202** instances where one demonstrably exists — 30 %, the number every later variant is measured against. Two findings concern method rather than results. First, **the chart changed underneath the measurement**: between two fetches three days apart, six of 273 tile keys received new content from the national mapping authority — up to 349 additional obstacle polygons and 225 additional point hazards per key — with no version, date, or signal in the tile API. Five baseline rows moved and one route changed outright. Standard benchmark discipline freezes code and instances; it rarely freezes *fetched third-party data*, and we argue it must. Second, and against ourselves: a manifest hash over the tile snapshot is necessary but *not sufficient*, because a tile can be present and empty, and the gate cannot tell the difference. We further document two cases where the measuring apparatus reproduced the very defect it was built to expose. We give the derivation of the 2.808 % error budget, the evidence that the verification checks have teeth (they caught four real defects, two of them inside the checker itself), and the coverage gaps we know the corpus has.

Keywords: experimental algorithmics; benchmark construction; reproducibility; data provenance; reference solutions; maritime route planning.

Sammendrag. Vi rapporterer byggingen og den uavhengige verifisering av et frosset måleapparat for maritim ruteplanlegging: et korpus på 273 instanser, en fasit med *målt* framfor antatt feilbudsjett, og en karakterisering av dagens planlegger mot begge. Hvert artefakt har kryptografisk identitet og en port som nekter kjøring ved avvik. Dagens planlegger svarer «ingen vei» på **61 av 202** instanser der vei beviselig finnes — 30 %, tallet alle senere varianter måles mot. To funn gjelder metode framfor resultat. For det første **endret kartet seg under målingen**: mellom to hentinger tre dager fra hverandre fikk seks av 273 flisnøkler nytt innhold fra Kartverket — opptil 349 nye hindringsflater og 225 nye punktfarer per nøkkel — uten versjon, dato eller signal i flis-API-et. Fem baseline-rader flyttet seg, og én rute ble en annen. Etablert benchmark-disiplin fryser kode og instanser; den fryser sjelden *hentede tredjepartsdata*, og vi argumenterer for at den må. For det andre, om prosjektets egen port: en manifest-hash over flis-snapshotet er nødvendig, men *ikke tilstrekkelig* — en flis kan være til stede og tom, og porten ser ikke forskjell. Vi dokumenterer i tillegg to tilfeller der måleapparatet gjentok nøyaktig den feilen det var bygget for å avdekke.

1 Innledning

Et ruteforslag til en fritidsbåt er rådgivende: skipperen skal lese ruta, forstå den, justere den, og deretter styre etter den [2]. Vi bygger en slik planlegger, og valget av arkitektur skal avgjøres eksperimentelt gjennom fire faser målt mot én felles referanse.

Da oppstår et krav som er lett å undervurdere: referansen må stå stille. Fire faser måles mot den, over måneder. Enhver inndata som endrer seg i mellomtiden gjør tallene usammenliknbare uten å gi noe signal om det — de ser fortsatt riktige ut.

Denne artikkelen rapporterer hvordan apparatet ble bygget og uavhengig verifisert, og to ting som gikk galt underveis. Begge gjelder metode framfor resultat, og begge er, tror vi, generelle.

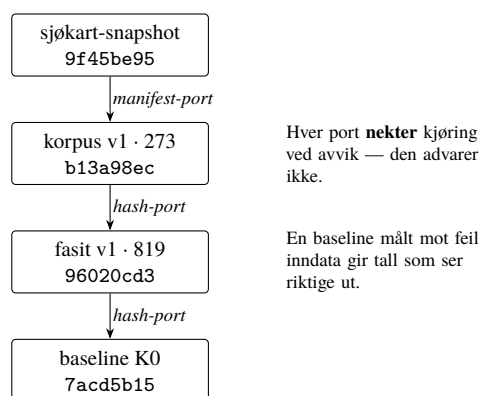
1.1 Bidrag

1. Et frosset måleapparat — korpus, fasit og baseline — der hvert artefakt har kryptografisk identitet og en port som *nekter* framfor å advare (§2–§4).

2. Et feilbudsjett som er *målt*, ikke antatt, med dokumentasjon av at gjetningen ble avvist av data (§3).
3. Dokumentasjon av en stille, additiv oppdatering av nasjonale sjøkartdata målt *in vivo*, med avtrykk ned på instansnivå (§5).
4. Et argument mot vår egen port: hash over et snapshot fanger ikke en flis som er til stede og tom (§6).
5. To tilfeller der måleapparatet gjentok feilen det målte (§7), og de dekningshullene vi vet korpuset har (§8).

2 Apparatet

Apparatet er en kjede av fire frosne artefakter (figur 1). Prinsippet gjennom hele kjeden er at porten *nekter* framfor å advare. En advarsel man kan klikke bort er ikke en port; en baseline målt mot feil korpus eller feil kart gir tall som ser riktige ut og ikke betyr noe, og de blir stående som referanse for fire faser.



Figur 1: Artefaktkjeden. Identiteten til hvert ledd er en SHA-256, og hvert ledd er portet mot leddet over. Porten som mangler i de fleste oppsett er den øverste: kartet.

2.1 Korpuset

Korpusbygging for rutenettsplanlegging har en etablert tradisjon [8]; det som skiller vårt tilfelle er at instansene ikke kan trekkes fra et gitt kart, men må konstrueres mot geometri som selv må hentes og frys.

273 instanser trukket stratifisert på farvannstype (fjord, indre skjærgård, åpen kyst) $\times$ strekningslengde (< 5 nm, $5\text{--}30$ nm, > 30 nm), 30 per stratum, pluss tre forankringsinstanser hvorav én er en faktisk navigert rute. Trekkingen er seedet; identiteten er konfigfila sammen med `instanshash b13a98ec`.

Verifiseringen re-implementerte alle kontroller uavhengig av generatorens kode og kjørte generatoren to ganger på en annen maskin dagen etter. Utvalgte tall:

- To kjøring byte-identiske; `instanshash` reproduisert eksakt på annen maskin og dag.
- 15 havne- og 15 sjøinstanser i *hvert* stratum — ikke bare 135/135 i sum.
- Alle 135 havneinstanser har nøyaktig ett endepunkt innenfor kystkonturen (125 på land, 10 på tørrfall) og ett i vann. Det vanskelige tilfellet er altså reelt til stede, ikke flyttet ut i trygt farvann.
- 0 instanser utenfor sitt distansebånd; største avvik mellom lagret og reberegnet luftlinje: **0,93 m**.
- Spredningsregelen ($\text{start--start} \geq 800$ m) holder i alle strata; målt minimum 822 m.

Verifiseringen fant også en svakhet som ikke rammet v1, men som kunne rammet en senere versjon: *en feilet flishenting ble klassifisert som «utenfor data», og kandidaten forkastet*. En forbigående nettverksfeil kunne dermed stille endre trekkingen. Begge kontrollkjøringene hadde null slike, så korpus v1 er upåvirket. Mekanismen er verdt å merke seg, for den er generatorens eget prinsipp — «fravær av data er ikke fravær av land» — anvendt feil vei på nettverket.

3 Fasiten: et målt feilbudsjett

Fasiten er en finoppløst referanseløsning, bundet til korpuset med en hash-port. 819 rader (273 instanser $\times$ tre dybdeprofiler), identitet 96020cd3.

Det interessante er ikke fasiten, men hvordan feilgrensen ble fastsatt. En rasterbasert fasit har en diskretiseringsfeil [5, 1], og den oppgis ofte som et rundt tall fra litteraturen. Vi krevde at den skulle utledes fra det faktiske oppsettet:

retningsledd (utledet, 2^5 retn., maks gap $18,435^\circ$)	1,3081 %
hindringsledd (målt)	1,5 %
sum	2,808 %

Retningsleddet følger av nabolagets form: for et 2^k -nabolag er det maksimale vinkelavviket per steg kjent i lukket form [7], og med 2^5 retninger gir det et største gap på $18,435^\circ$. Det leddet er ren geometri, og lot seg utlede uten måling.

Hindringsleddet er poenget. Det ble først *gjettet* til 0,7 %. Konvergenssjekken avviste gjetningen — den målte 1,351 % i tettete skjærgård — cellene ble halvert til 2,5 m for de korteste instansene, og budsjettet satt fra data.

Summen på 2,808 % overstiger det veiledende målet på omtrent 2 %. Den ble likevel akseptert, og begrunnelsen er registrert: tallet er målt og ikke valgt, empirisk stramming er maksimalt 1,384 %, margin til løsningsmålets 5 % er fortsatt $1,8\times$, og feilen er *ensidig* — fasiten er aldri kortere enn sant optimum. Ensidigheten er ikke tilfeldig: diskretisering av et kontinuerlig baneproblem gir en systematisk skjevhet i én retning [9], og det er den retningen man vil ha den i når en falsk «klar» kan sette båten på grunn.

Et budsjett man har målt seg til og som er for stort, er mer brukbart enn et man har valgt seg til og som ser pent ut.

3.1 Sjekker med tenner

En verifikasjon som aldri finner noe er ikke bevis på kvalitet; den er oftest bevis på at sjekkene er for svake. Fasitens innebygde kontroller — 618 nedre skranker, 618 kollisjonssjekker mot u-erodert geometri, 51 symmetriprøver, 30 konvergensprøver — rapporterte null brudd. At de likevel har tenner, er demonstrert av at de fanget fire reelle feil underveis, og **to av dem satt i kontrolløren selv**, fanget av kontrollørens egne tester.

Forskningssporet la til en uavhengig kollisjonskontroll med egen geometrikode og egen flislesing, som ikke deler noe med fasitens kryssjekk: 14 stratifisert utvalgte ruter, 8 845 punktprøver, hvert ≤ 25 m. Minste målte klaring 53,86 m, ingen under 49,9 m — og hver rute målte $+1,95$ til $+7,54$ m *bedre* enn fasitens egen rapporterte minimumsklaring. Den konservative rapporteringen underlover altså konsekvent, aldri motsatt. Det er den retningen man vil ha feilen i når en falsk «klar» kan sette båten på grunn.

En dokumentert egenskap, ikke en feil, kom også ut av dette: i to instanser mangler en *grunnere* profil en vei som en dypere har. Mekanismen er bekreftet i dataene — den

grunne profilen snapper kortere (74,66 m mot 90,43 m) og havner i en fri lomme uten utgang.

4 Baselinen

Dagens planlegger — uniformt rutenett, A^* , geometribevisst etterforenkling — er frosset som `k0-baseline` og kjørt på hele korpuset mot fasiten. Det er referansen alle senere varianter måles mot, og hovedtallet er:

61 av 202 instanser der vei beviselig finnes, får svaret «ingen vei». 30 %.

Det er ikke en marginal svakhet. Det er nesten hver tredje spørring som feiler åpent, og det er grunnen til at prosjektet finnes.

5 Kartet endret seg under målingen

Forskningssporets kalde fullkjøring med ferske fliser dagen etter leveransen ga en annen radhash enn leveransen. Diagnosen er entydig:

- **Seks av 273 flisnøkler fikk nytt innhold** mellom hentingene 27. og 28. juli — alle i indre skjærgård.
- Oppdateringen er **additiv og usignalisert**: opptil **+349 hindringsflater** og **+225 punktfarer** per nøkkel (land, tørrfall, gruntområder, grunner, skjær). Ingen versjon, ingen dato, intet signal i flis-API-et — bare andre tall.
- **Fem baseline-rader flyttet seg**. På én instans endret selve ruta seg, og raden ble et invariantbrudd. På en annen driftet inndata *uten* at raden endret seg — driften var der, men usynlig i utdataene.

Leveransens tall var internt konsistente: fasit og harness var kjørt på samme kart-æra. Problemet er ikke at noen målte feil. Problemet er at ingen *kunne visst* at kartet hadde endret seg, fordi datakilden ikke sier fra.

Konsekvensen ble at flis-snapshotet er frosset på samme vilkår som korpus og fasit: manifest-hash over den sorterte lista av innholdshasher per nøkkel, port som nekter kjøring ved avvik, og diagnose på nøkkelnivå framfor bare en dom. Snapshotet (11 GB) arkiveres utenfor versjonskontroll; manifestet er committet, så en fremmed kopi kan verifiseres fil for fil uten å ha vår disk.

Diagnose på nøkkelnivå var ikke pynt. Det var nettopp den som lot oss skille «Kartverket har oppdatert kartet» fra «harnessen er i stykker» — to funn som ser helt like ut i en hash-sammenlikning.

5.1 Hvorfor dette generaliserer

Etablert disiplin i eksperimentell algoritmikk [4, 6] fryser kode og instanser. Den sier lite om inndata som *hentes* fra en tredjepart i måleøyeblikket — kartfliser, geokodingstjenester, høydemodeller, trafikkdata. Slike kilder oppdateres kontinuerlig og som regel uten versjonering, fordi de er bygget for bruk og ikke for måling.

Vi observerte en endring på 2,2 % av nøklene i løpet av ett døgn. Et eksperiment som går over måneder, mot samme kilde, uten port, vil med stor sannsynlighet måle to ulike verdener og rapportere forskjellen som en algoritmeeffekt.

6 Hva porten ikke fanger

Her må vi argumentere mot vårt eget tiltak.

Manifest-porten sikrer at flisene er de samme. Den sikrer *ikke* at flisene inneholder noe. En flis kan være til stede og tom, og hashen er da like fornøyd som om den var full av geometri.

Det er ikke hypotetisk. Vi har målt et feltilfelle der planleggeren bygde rutenettet uten hindringer i det og meldte en rett strek over land som klar [3]. Da vi kontrollerte baselinen for samme signatur, fant vi 58 av 273 rader som er rette streker meldt klar med lengde lik luftlinja. Vår første frikjenning av dem lød at en manglende flis ville felt manifest-porten. **Den begrunnelsen holder ikke**, nettopp fordi en flis kan være til stede og tom.

De 58 er reelle, men av en annen grunn: harnessen sender med et dekningsområde, celler utenfor det er *sperret* og ikke frie, og hvert legg må i tillegg bestå en dekningssekk. Fri celle betyr da «lest og farbar». Det er dekningsmekanismen som frikjenner radene — ikke porten.

Lærdommen er generell nok til å formuleres: *en integritetsport svarer på om inndata er de samme, aldri på om de er tilstrekkelige*. De to spørsmålene forveksles lett, fordi den samme hashen brukes til å stille dem begge.

7 Når apparatet gjentar feilen det måler

To ganger under fase 0 gjentok måleapparatet formen på nøyaktig den feilen det var bygget for å avdekke. Vi tror ikke det er tilfeldig.

Første gang. Planleggerens etterforenkling godtar snarveier som ikke *krysser* noe. «Krysser» og «er nær» er ulike predikater: flatene har ikke noe avstandsbegrep i verifikatoren, så klaringskravet håndheves bare mot punktfarer. Vi kvantifiserte dette på korpuset og fant en rute som passerer **2,11 m fra land** med klar: `true` — med 493 m åpent vann på motsatt side.

Men målingen som skulle vise dette, målte en annen polylinje enn den løftet gjelder: den utelot endepunkt-leggene. 2,11-metersgrazingen lå på et endepunkt-legg og var derfor usynlig i leveransens egne tall. Målingen av et blindfelt hadde sitt eget blindfelt, av samme form.

Andre gang. I perturbasjonsstudien [3] ble ruteendring først målt med en hash som inkluderte startpunktet — altså nøyaktig det som ble perturbert. Feilen ble oppdaget og rettet. Den uavhengige verifiseringen fant deretter samme feilklasse én gang til i det rettede arbeidet: stiendring ble målt på en hash som ser flyttallspreisjon, og talte ni kjøring som «flyttet» der punktene avvek under 5 mm.

Fellesnevneren er at et *identitetsmål* ble brukt til å svare på et *avstandsspørsmål*. Vi foreslår regelen som konvensjon der artefakter fryses med hash: **hash for identitet, metrikk**

for forflytning — aldri hash for forflytning.

8 Hva korpuset ikke dekker

Et frosset apparat er ikke et gyldig apparat. To dekningshull er funnet i ettertid, og begge gjelder artefaktene denne artikkelen beskriver.

Korpuset underrepresenterer sitt eget kvalitetsmål. Blant løsningsmålene er et krav om leggvastand: ruta skal ikke bestå av så korte legg at en rormann ikke kan handle på dem. Målt over korpuset er median legglengde 9 272 m, og legg under 50 m forekommer i bare 25 av 273 instanser. Målt i felt, i trangt farvann, er den samme størrelsen 57 m med 47 % av leggene under 50 m. Ikke en motsigelse — ulike populasjoner — men korpuset samler nesten ikke de forholdene der målet faktisk svikter. Det er verre enn en tilfeldig svakhet, fordi det samvarierer: i det samme trange farvannet er ruta også minst stabil [3].

Vi endrer ikke korpuset. Det er frosset, og en omsampling ville gjort alle arbeidspakkene usammenliknbare. Truselen håndteres ved at senere faser må rapportere per stratum framfor aggregat, og ved supplerende instrumenter.

Et felt måler ikke det navnet lover. Harnessens klassifiserteCeller er *tabellstørrelsen* (kolonner $\times$ rader), ikke antall celler der arbeid ble utført. Tabelltaken er seks ganger celletaket, og 87,5 % av instansene har en tabell større enn celletaket. Feltnmåling viser hvor mye det betyr: to kjøring med 43 524 celler brukte 831 ms, mens en kjøring med 46 360 celler — seks prosent *flere* — brukte 10 407 ms. Kostnaden drives av hindringstetthet, ikke av celletall.

Generaliseringen er at et stratifisert korpus balanserer det som er lett å stratifisere på, og ikke nødvendigvis det kvalitetsmålene er følsomme for. Vi anbefaler at benchmark-korpus rapporterer dekningen av *hvert* måls kritiske område, ikke bare stratumbalansen.

9 Anbefalinger

1. **Frys hentede tredjepartsdata som et artefakt.** Manifest over innholdshasher, port som nekker, diagnose på nøkkelnivå. Kode og instanser er ikke nok når inndata hentes.
2. **Mål feilbudsjetter, ikke siter dem.** Og oppgi når målingen avviste anslaget — det er den opplysningen som gjør budsjettet troverdig.
3. **Krev at kontrollene har tenner.** Rapporter hva de fanger, inkludert feil i kontrollene selv.
4. **Hash for identitet, metrikk for forflytning.**
5. **Rapporter dekningen av hvert måls kritiske område,** ikke bare stratumbalansen.
6. **Oppgi grensen for din egen port.** En integritetsport svarer på om inndata er de samme, aldri på om de er tilstrekkelige.

Tilgjengelighet

Alle artefakter er frosset med SHA-256 og oppgitt i artefaktens hoder: korpus b13a98ec, fasit 96020cd3, baseline 7acd5b15, flismanifest 9f45be95 (og 1dab3d0a, som dokumenterer driften). Sjøkartdata er Kartverkets, tilgjengeliggjort under NLOD. Verifikasjonsrapportene for hver arbeidspakke er skrevet av et forskningsspor som er uavhengig av implementasjonen.

Erklæringer

Interessekonflikt: Forfatteren eier Fellag AS, som utvikler BoatNav. Artikkelen måler Fellags eget produkt.

KI-assistanse: Arbeidet er utført med gjennomgående assistanse fra Claude (Anthropic) — i analysen, i verifisering og i manuskriptet. Artefaktene er bygget av ett spor og verifisert uavhengig av et annet, med egne kontrollimplementasjoner. Argumentet mot prosjektets egen port i §6 og selvreferansen i §7 er verifikasjonsspoets funn. Alle måletall stammer fra BoatNav-implementasjonen og de frosne artefaktene. Forfatteren står ansvarlig for innholdet.

Bidragsroller (CRediT): Conceptualization, Methodology, Software, Investigation, Formal analysis, Validation, Visualization og Writing er utført av forfatteren med KI-assistanse som beskrevet over. *Validation* er holdt uavhengig av implementeringen: verifikasjonen reimplemterer kontrollene framfor å gjenbruke dem.

Data og kode: Artefaktene er frosset med SHA-256 (korpus b13a98ec, fasit 96020cd3, baseline 7acd5b15, flismanifest 9f45be95) og oppgitt i artefaktens hoder. Målingene og eksperiment-harnessen planlegges åpnet sammen med resultatartikkelen; inntil da tilgjengelig på forespørsel. Sjøkartdata er Kartverkets, under NLOD.

Ansvarsavgrensning: BoatNav gir veiledende rutforslag for fritidsbruk og er ikke et godkjent navigasjonshjelpemiddel.

Versjon 1.3 (1. august 2026): to formuleringer skrevet om etter registerporten (B8): «mot oss selv» og «nettopp derfor» er erstattet av beskrivelser av hva som faktisk ble funnet. Ingen tall eller konklusjoner er endret.

Versjon 1.2 (30. juli 2026): erklæringer om interessekonflikt, KI-assistanse og bidragsroller lagt til etter fagfelleprotokollen.

Versjon 1.1 (30. juli 2026): referanser som sto ubrukt i litteraturlista er knyttet til teksten der innholdet hviler på dem. Ingen påstander, tall eller konklusjoner er endret i noen av versjonene.

Referanser

- [1] J. Bailey *mfl.* Path-planning on grids: the effect of vertex placement on path quality. *AIIDE*, 2011.
- [2] O. J. M. Iversen. Automatisk ruteplanlegging for fritidsbåter i norsk skjærgård: problemkarakterisering, målinger og forskningsagenda. *BoatNav Research Note 2026-01*, Fellag AS, 2026.
- [3] O. J. M. Iversen. Rutenettet er ankret til spørningen, ikke til verden: startpunktsensitivitet i mobil ruteplanlegging for skjærgård. *BoatNav Research Note 2026-02*, Fellag AS, 2026.
- [4] D. S. Johnson. A Theoretician's Guide to the Experimental Analysis of Algorithms. I *Data Structures, Near Neighbor Searches, and Methodology*, AMS, 2002.
- [5] S. Kramm *mfl.* On the effect of grid discretization on path length. *Symposium on Combinatorial Search (SoCS)*, 2018.
- [6] C. C. McGeoch. *A Guide to Experimental Algorithmics*. Cambridge University Press, 2012.
- [7] N. Rivera, C. Hernández, N. Hormazábal og J. A. Baier. The 2^k Neighborhoods for Grid Path Planning. *Journal of Artificial Intelligence Research* 67, 2020.
- [8] N. Sturtevant. Benchmarks for Grid-Based Pathfinding. *IEEE Transactions on Computational Intelligence and AI in Games* 4(2), 2012.
- [9] J. N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control* 40(9), 1995.