معمارية المستكشف-الملاح التكيفية للوكلاء الذاتيين

E-NAV: Explorer–Navigator Adaptive Agent Architecture

إطار نظري وهندسي للكارتوغرافيا الديناميكية، الاستكشاف المقيد، الذاكرة التجريبية، وإعادة التخطيط المرن في أنظمة الذكاء الاصطناعي الوكيلية

الأطروحة المركزية

تقترح هذه الورقة أن الوكيل المتقدم لا يحتاج إلى «خريطة كاملة» قبل بدء العمل؛ بل يحتاج إلى آلية منضبطة لإنتاج الخريطة أثناء التنفيذ، قياس عدم اليقين، اختبار الفرضيات محلياً، تسجيل الخبرة، وإعادة توجيه المسار مع الحفاظ على الهدف والقيود. يتحول بذلك نموذج «المستكشف/الملاح» من استعارة إلى بنية تحكم قابلة للقياس والاختبار.

نسخة بحثية أولية - أغسطس 2026

نوع الورقة: Conceptual / Architectural Research Paper

الملخص

تواجه الأنظمة الوكيلية المبنية على النماذج اللغوية الكبيرة تحدياً بنيوياً: فهي مطالبة بالعمل في بيئات مفتوحة، متغيرة، وغير مكتملة المعرفة، بينما تفترض كثير من مسارات التنفيذ ضمنياً أن الخطة الأولية، وصف الأدوات، أو تمثيل البيئة كافٍ للوصول إلى الهدف. تُظهر أدبيات ReAct والتفكير الشجري والتأمل اللفظي والتعلم التجريبي والوكلاء المتجسدين أن المزج بين الاستدلال والفعل، التفرع والعودة إلى الخلف، الذاكرة، واستخراج الدروس من المسارات السابقة يحسن القدرة على حل المهام. كما تكشف بيئات التقييم الواقعية مثل WebArena وOSWorld وWindowsWorld أن الانتقال من مهام قصيرة ومعزولة إلى سير عمل طويل ومتعدد التطبيقات لا يزال يسبب انخفاضاً كبيراً في الموثوقية والكفاءة. [1–10، 14–17]

تقترح هذه الورقة إطار E-NAV، أو «معمارية المستكشف-الملاح التكيفية»، بوصفه بنية تحكم للوكلاء الذاتيين في البيئات غير المكتملة. يقوم الإطار على سبعة مكونات مترابطة: بوصلة المهمة، محرك الكارتوغرافيا الاعتقادية، طبقة بعثات الاستطلاع، مخطط المسار، ذاكرة سجل الرحلة، محرك الصلابة وإعادة التوجيه، وحلقة التحقق من الأثر في العالم. ويُصاغ اختيار الفعل كمسألة تحسين متعددة المعايير توازن بين التقدم نحو الهدف، المكسب المعلوماتي، قابلية الاسترداد، قيمة المعرفة القابلة لإعادة الاستخدام، الكلفة، المخاطر، وخطر الانحراف غير القابل للعكس. كما تُعرّف الورقة مقاييس قابلة للاختبار مثل «مكسب الكارتوغرافيا»، «مرونة الاسترداد»، «عائد إعادة استخدام سجل الرحلة»، و«معدل تكرار الفشل».

الورقة نظرية وهندسية ولا تدّعي نتائج تجريبية جديدة؛ بدلاً من ذلك تقدم فرضيات قابلة للتكذيب وبروتوكول تقييم مقترح على AgentBench وWebArena وOSWorld وOSWorld-MCP وWindowsWorld. الفرضية الأساسية هي أن فصل «ثبات المهمة» عن «لدونة المسار»، وإدخال قيمة معلوماتية صريحة للاستكشاف، وتحويل الأخطاء المتكررة إلى حواجز ومهارات قابلة لإعادة الاستخدام، يمكن أن يرفع موثوقية الوكلاء في المهام طويلة الأفق مع خفض الفشل المتكرر والهدر في الأدوات والرموز.

الكلمات المفتاحية: Agentic AI، Autonomous Agents، Adaptive Planning، Dynamic Cartography، Exploration، Experiential Memory، Reflection، Tool Use، World Models، Resilience، Replanning، Active Inference، Computer-Use Agents.

Abstract

This paper proposes E-NAV, an Explorer–Navigator Adaptive Agent Architecture for autonomous agents operating under incomplete and changing world knowledge. Rather than treating the environment map or initial plan as fixed, E-NAV models execution as continual cartography: probe, observe, update beliefs, re-plan, log experience, and verify world effects. The architecture combines a Mission Compass, Belief Cartography Engine, Scout/Probe Layer, Route Planner, Operational Logbook Memory, Resilience & Rerouting Engine, and World-Effect Verification Loop. A multi-objective action criterion balances goal progress, expected information gain, recoverability, reusable knowledge value, computational/tool cost, risk, and irreversibility. The paper

defines falsifiable hypotheses and an evaluation protocol across interactive agent benchmarks. It is a conceptual/architectural contribution and does not claim new empirical results.

خريطة الورقة

- 1. المقدمة والسياق البحثي
- 2. مشكلة البحث وأسئلته ومساهماته
- 3. المنهجية ونطاق الادعاء العلمي
- 4. الأساس النظري والأعمال ذات الصلة
- 5. الفجوة العلمية: من الخطة الثابتة إلى الكارتوغرافيا المستمرة
- 6. المبادئ التنفيذية لنموذج المستكشف-الملاح
- 7. معمارية E-NAV ذات الطبقات السبع
- 8. الصياغة الرياضية ودالة القرار
- 9. الذاكرة وسجل الرحلة وتحويل الفشل إلى معرفة
- 10. الاستكشاف المقيد وبعثات الاستطلاع
- 11. إعادة التخطيط والصلابة تحت الاضطراب
- 12. التحقق من الأثر والحوكمة والسلامة
- 13. الخوارزمية التشغيلية المقترحة
- 14. بروتوكول التقييم والتجارب
- 15. دراسات الإزالة والفرضيات القابلة للتكذيب
- 16. تطبيقات هندسية وريادية وبحثية
- 17. المناقشة والحدود
- 18. برنامج البحث المستقبلي
- 19. الخاتمة
- المراجع
- الملحق أ: تعريف المتغيرات والمقاييس
- الملحق ب: نموذج سجل الرحلة

1. المقدمة والسياق البحثي

انتقل الذكاء الاصطناعي التوليدي خلال الأعوام الأخيرة من نمط «الاستجابة للنص» إلى نمط أكثر فاعلية تتصرف فيه النماذج كمتحكمات معرفية تستقبل هدفاً، تستدعي أدوات، تراقب بيئة خارجية، وتعدّل سلوكها عبر سلسلة طويلة من القرارات. هذه النقلة لا تعني أن النموذج أصبح وكيلاً موثوقاً تلقائياً؛ فالوكالة العملية تتطلب إدارة الحالة، الذاكرة، اختيار الأدوات، التحقق من النتائج، والتعامل مع عدم اليقين والانقطاعات. بيّن ReAct أن تداخل الاستدلال والفعل يسمح بتحديث الخطة عبر الملاحظات بدلاً من إبقائها مساراً لغوياً مغلقاً، بينما قدّم Tree of Thoughts فكرة البحث عبر مسارات بديلة مع التقييم والعودة إلى الخلف. [1، 2]

في المقابل، ركزت Reflexion وExpeL ثم أعمال أحدث في التعلم التجريبي على السؤال التالي: ماذا ينبغي أن يفعل الوكيل بما حدث له؟ بدلاً من تجاهل المسار بعد انتهاء المهمة، يمكن ضغط التجربة إلى تأملات أو إرشادات أو معرفة قابلة للاسترجاع. أظهر Reflexion أن التغذية الراجعة اللفظية والذاكرة العرضية يمكن أن تحسن المحاولات التالية دون تحديث أوزان النموذج، ووسّع ExpeL هذا التصور إلى استخراج خبرة قابلة للنقل بين المهام. وفي 2026، طرحت أعمال مثل Experiential Reflective Learning وOnline Experiential Learning صيغاً أكثر منهجية لاستخراج معرفة قابلة للنقل من مسارات النشر ثم إعادة استخدامها أو ترسيخها. [3، 6، 15، 16]

لكن هذه التطورات لا تلغي المشكلة الأساسية: في البيئة الواقعية لا تكون «الخريطة» معروفة بالكامل. تتغير الواجهات، تفشل الأدوات، تختلف صلاحيات الحسابات، تتبدل استجابات APIs، وقد يكون وصف المهمة نفسه ناقصاً. لذلك لا يكفي التخطيط على خريطة ثابتة أو استخدام الذاكرة كملحق. المطلوب هو اعتبار تمثيل البيئة ذاته كائناً ديناميكياً يخضع للتحديث المستمر، وأن يصبح الاستكشاف المنضبط جزءاً من دالة القرار، لا سلوكاً طارئاً عند الفشل.

تكشف بيئات القياس هذه الفجوة بوضوح. اختبرا AgentBench وWebArena الوكلاء في بيئات تفاعلية مفتوحة نسبياً، ثم نقل OSWorld التقييم إلى تطبيقات وأنظمة تشغيل حقيقية متعددة الوسائط. وفي 2025 أضاف OSWorld-MCP بعداً مهماً: الوكيل قد يمتلك واجهة GUI وأدوات بروتوكولية في الوقت نفسه، لكن امتلاك الأداة لا يضمن اختيارها أو استخدامها جيداً. وفي 2026 قدّم WindowsWorld مهاماً مهنية متعددة التطبيقات ومتعددة الأهداف الجزئية، وأظهر أن النجاح ينخفض بشدة عندما تصبح المهام مشروطة وتستلزم تنسيقاً بين ثلاثة تطبيقات أو أكثر. [8–10، 14، 17]

سؤال الورقة

المشكلة ليست فقط «كيف نخطط؟» بل «كيف نخطط عندما تكون الخريطة ناقصة، وكيف نجعل كل فعل يساهم في تحسين الخريطة نفسها، وكيف نمنع المعرفة الجديدة من الضياع بعد انتهاء المهمة؟»

2. مشكلة البحث وأسئلته ومساهماته

2.1 مشكلة البحث

تفترض كثير من معمارية الوكلاء مساراً ضمنياً من النوع: هدف ← تخطيط ← تنفيذ ← ملاحظة ← تصحيح. إلا أن هذا المسار قد يفشل في تمثيل حقيقة أن «المعرفة بالبيئة» و«الخطة» و«قابلية الاسترداد» تتغير في أثناء التنفيذ. حين يكتشف الوكيل أن API غير متاح، أو أن حقلاً في واجهة الرسوم تغير، أو أن إجراءً يتطلب مصادقة بشرية، فالمشكلة ليست مجرد خطأ في خطوة واحدة؛ بل حدث كارتوغرافي يغيّر النموذج التشغيلي للمجال.

تعرّف هذه الورقة المشكلة رسمياً بوصفها اتخاذ قرار طويل الأفق تحت خريطة اعتقادية جزئية ومتغيرة، مع ضرورة الحفاظ على الهدف والقيود، تعظيم التعلم من الأفعال، وتقليل الإجراءات غير القابلة للعكس. وهذا يختلف عن التخطيط التقليدي على نموذج انتقال معلوم، كما يختلف عن الاستكشاف غير المقيد الذي قد يهدر الموارد أو يخلق مخاطر.

2.2 أسئلة البحث

1. كيف يمكن تمثيل «الخريطة التشغيلية» للوكيل كحالة اعتقادية ديناميكية بدلاً من مجموعة تعليمات ثابتة؟
2. كيف يمكن إدخال القيمة المعلوماتية للفعل ضمن قرار الوكيل إلى جانب التقدم المباشر نحو الهدف؟
3. كيف يمكن لآلية سجل الرحلة أن تحوّل المسارات الناجحة والفاشلة إلى معرفة قابلة لإعادة الاستخدام، مع تجنب تضخم السياق؟
4. كيف يمكن قياس الصلابة ليس فقط بمعدل النجاح النهائي، بل بقدرة الوكيل على التعافي من اضطراب غير متوقع مع كلفة إضافية محدودة؟
5. ما التجارب التي يمكن أن تكذب فرضية أن معمارية E-NAV أفضل من خطوط أساس تعتمد على التخطيط/التأمل دون كارتوغرافيا صريحة؟

2.3 المساهمات

- صياغة مفهوم «الكارتوغرافيا أثناء المسير» كآلية تحكم للوكيل: كل تنفيذ إما يحقق الهدف أو يقلل عدم اليقين أو يفعل الاثنين معاً.
- اقتراح معمارية E-NAV من سبع طبقات تفصل بوضوح بين بوصلة المهمة، خريطة الاعتقاد، الاستطلاع، التخطيط، الذاكرة، الاسترداد، والتحقق من الأثر.
- تقديم دالة قرار متعددة المعايير ومجموعة مقاييس جديدة قابلة للقياس، من بينها Cartographic Gain وRecovery Elasticity وLogbook Reuse Yield وRepeat Failure Rate.
- اقتراح بروتوكول «تحويل الفشل إلى قدرة» حيث تُرقّى أنماط الفشل المتكررة إلى قواعد وقائية واختبارات ومهارات بعد اجتياز تحقق تجريبي.
- وضع بروتوكول تجريبي ودراسات إزالة تسمح بفصل أثر كل مكوّن وتحديد ما إذا كانت الزيادة في النجاح تأتي من الاستكشاف أو الذاكرة أو إعادة التخطيط أو مجرد زيادة ميزانية الرموز والأدوات.

3. المنهجية ونطاق الادعاء العلمي

تتبع الورقة منهجية بحث مفاهيمي-معماري مدعومة بمراجعة انتقائية للأعمال الأولية ذات الصلة. لا تقدم الورقة نتائج أداء جديدة ولا تنسب إلى E-NAV تفوقاً تجريبياً قبل الاختبار. ولذلك تُصاغ الاستنتاجات على هيئة: (أ) مبادئ مشتقة من فجوات معروفة في الوكلاء الحاليين، (ب) تصميم معماري مقترح، (ج) تعريفات رياضية، و(د) فرضيات قابلة للتكذيب وخطة تجارب.

تم اختيار الأعمال المرجعية لأنها تمثل مكونات مباشرة في المشكلة: ReAct للتداخل بين الاستدلال والفعل؛ Tree of Thoughts للبحث والعودة؛ Reflexion وExpeL وERL/OEL للتعلم من الخبرة؛ Voyager للاستكشاف طويل المدى ومكتبة المهارات؛ active inference لدمج القيمة الأداتية والإبستيمية؛ AgentBench/WebArena/OSWorld/OSWorld-MCP/WindowsWorld لقياس السلوك في بيئات تفاعلية متصاعدة الواقعية؛ وMIRROR للتأمل قبل التنفيذ وبعده في سياق استخدام الأدوات. [1–17]

حدود المنهجية مهمة: مصطلح «الكارتوغرافيا» هنا استعارة هندسية ذات تعريف رسمي، وليس ادعاءً بأن الوكيل يبني خريطة جغرافية. الخريطة قد تكون رسم اعتماد للأدوات، مخطط حالات GUI، نموذج صلاحيات، ذاكرة حقائق تشغيلية، توزيع احتمال على نتائج الأفعال، أو مزيجاً منها. كما أن «المستكشف» لا يعني الاستقلال غير المقيد؛ بل الاستكشاف مشروط بميزانية ومخاطر وحدود صلاحيات واضحة.

4. الأساس النظري والأعمال ذات الصلة

4.1 الاستدلال والفعل والتخطيط متعدد المسارات

قدّم ReAct نموذجاً أساسياً لتداخل آثار الاستدلال مع الأفعال والملاحظات. هذه البنية مهمة لـ E-NAV لأنها تجعل البيئة مصدراً للمعرفة أثناء التفكير، لا مجرد مسرح لتنفيذ خطة مكتملة سلفاً. لكن E-NAV يضيف فوق ذلك تمثيلاً صريحاً لما تغيّر في «الخريطة» بعد كل ملاحظة، ويعامل قيمة المعلومة كجزء من اختيار الفعل. [1]

أما Tree of Thoughts فيتجاوز المسار الأحادي عبر إنشاء وحدات تفكير قابلة للتقييم والبحث والعودة إلى الخلف. تتقاطع هذه الفكرة مع «الملاحة» في E-NAV، لكنها تظل في جوهرها إطاراً للبحث في فضاء الأفكار. في النظام المقترح، قد تكون العقد مسارات تنفيذ حقيقية، وحالة البيئة نفسها تتغير نتيجة الأفعال، لذلك يجب أن يرتبط التفرع بتقدير عدم القابلية للعكس وكلفة الاستطلاع. [2]

4.2 التأمل والذاكرة التجريبية

Reflexion يوضح أن الفشل لا يحتاج دائماً إلى تحديث أوزان النموذج كي يصبح مفيداً؛ يمكن تحويله إلى وصف لغوي مخزن في ذاكرة عرضية يؤثر في القرار اللاحق. ExpeL يذهب أبعد إلى استخلاص خبرة مشتركة من مسارات متعددة. الأعمال الأحدث مثل ERL تركز على استخراج heuristics قابلة للنقل واسترجاع الأكثر صلة بالمهمة، بينما يقترح OEL دورة تعلم مستمرة تُستخرج فيها المعرفة من خبرة النشر ثم تُرسخ لاحقاً في النموذج. هذه النتائج تدعم مبدأ «سجل الرحلة»، لكنها تترك سؤالاً تصميمياً: ما الذي يجب تسجيله، وما الذي يستحق الترقية إلى قاعدة مستقرة، وكيف نربط السجل بالخريطة والصلابة؟ [3، 6، 15، 16]

4.3 الاستكشاف المفتوح ومكتبات المهارات

Voyager مثال مهم على وكيل طويل الأفق يستخدم منهجاً آلياً لاستكشاف بيئة Minecraft، مع مكتبة مهارات تنمو عبر الخبرة. القيمة النظرية هنا ليست البيئة ذاتها، بل المبدأ القائل إن الاستكشاف يمكن أن ينتج مخزوناً قابلاً لإعادة الاستخدام بدلاً من سلسلة محاولات منفصلة. E-NAV يعمم هذا المبدأ ليشمل بيئات الحاسوب والأدوات والأنظمة البرمجية، ويضيف شرطاً بأن المهارة المشتقة من الفشل لا تُعتمد إلا بعد التحقق من قابليتها للتكرار وحدود صلاحيتها. [5]

4.4 الاستكشاف الإبستيمي و**Active Inference**

توفر active inference أساساً نظرياً للتوفيق بين الوصول إلى نتائج مرغوبة وتقليل عدم اليقين. في هذا المنظور لا يكون الاستكشاف مكافأة جانبية، بل قيمة إبستيمية يمكن أن تدخل في اختيار السياسة. أعمال حديثة أعادت صياغة التخطيط القائم على Expected Free Energy بوصفه استدلالاً تغايرياً، مع مراعاة التعقيد والموارد المحدودة. يستلهم E-NAV هذه الفكرة دون الادعاء بأنه تطبيق كامل لـ active inference؛ فهو يستخدم نسخة هندسية مبسطة من «مكسب المعلومات المتوقع» ضمن دالة قرار عملية للوكلاء المبنيين على LLMs. [11، 12]

4.5 قياس الوكلاء في البيئات الواقعية

تظهر المقاييس التفاعلية أن الموثوقية تتدهور كلما زادت واقعية البيئة وطول الأفق. AgentBench يقيس الوكلاء عبر عدة بيئات، WebArena يركز على الويب الواقعي، وOSWorld يقدم مهاماً حقيقية على أنظمة تشغيل وتطبيقات متعددة. OSWorld-MCP يفصل بين استخدام GUI واستدعاء الأدوات عبر MCP، موضحاً أن توفر الأدوات لا يعني استخدامها بكفاءة. WindowsWorld يدفع نحو سير عمل مهني متعدد التطبيقات، ويبلغ عن معدلات نجاح تقل عن 21% على المهام متعددة التطبيقات لدى الوكلاء الذين اختبرهم، مع فشل ملحوظ في الحكم الشرطي والتنسيق عبر ثلاثة تطبيقات أو أكثر. هذه النتائج تجعل «إعادة التوجيه تحت الاضطراب» مقياساً مركزياً لا ملحقاً. [7–10، 14، 17]

العمل	المساهمة الأساسية	ما الذي يستفيده **E-NAV**	الفجوة المتبقية
ReAct [1]	تداخل الاستدلال والفعل	حلقة فعل-ملاحظة-تعديل	لا توجد خريطة اعتقادية/قيمة معلومات صريحة
Tree of Thoughts [2]	بحث متعدد المسارات وعودة	تفرع ومسارات بديلة	لا يعالج بالكامل أثر الأفعال غير القابلة للعكس
Reflexion [3]	تأمل لفظي وذاكرة عرضية	تحويل الفشل إلى إرشاد	لا يعرّف ترقية منهجية إلى guardrail/skill
Voyager [5]	استكشاف مفتوح ومكتبة مهارات	المهارات كتراكم خبرة	بيئة خاصة وتحتاج تعميماً على أدوات رقمية

الفجوة المتبقية	ما الذي يستفيده E-NAV	المساهمة الأساسية	العمل
الحاجة لربط الذاكرة بالخريطة والصلابة	سجل رحلة قابل للاسترجاع/التعلم	استخراج معرفة من الخبرة	ExpeL / ERL / OEL [6,15,16]
لا تقترح بذاتها معمارية التحكم المطلوبة	منصة قياس للتعافي وإعادة المسار	اختبار حاسوبي واقعي طويل	OSWorld / WindowsWorld [10,17]

5. الفجوة العلمية: من الخطة الثابتة إلى الكارتوغرافيا المستمرة

الفرق بين «التخطيط» و«الكارتوغرافيا» جوهري. التخطيط يختار مساراً بالنظر إلى نموذج ما؛ أما الكارتوغرافيا فتحدّث النموذج نفسه. عندما يكون نموذج العالم ناقصاً، فإن أفضل مسار ليس بالضرورة أقصر مسار نحو الهدف، لأن بعض الأفعال تكشف معلومات تقلل أخطاء لاحقة. من هنا يظهر مفهوم «فعل الاستطلاع» Probe Action: فعل منخفض المخاطر نسبيًا هدفه الأساسي التحقق من فرضية تشغيلية، مثل اختبار صلاحية API بعملية قراءة غير مدمرة قبل تنفيذ كتابة، أو فتح صفحة إعدادات لتحديد موضع عنصر قبل بدء سلسلة نقرات طويلة.

الفجوة الثانية هي أن الذاكرة غالباً تُستخدم كمخزن نصي، بينما يحتاج الوكيل إلى معرفة ما إذا كانت الخبرة محلية أو قابلة للتعميم. سجل رحلة جيد لا يقول فقط «فشل استدعاء X»، بل يميز: السبب، السياق، نسخة الأداة، شروط التكرار، مسار الاسترداد، ودرجة الثقة. دون ذلك تصبح الذاكرة مصدراً جديداً للهلوسة التشغيلية.

الفجوة الثالثة هي ضعف القياس المتعلق بالتعافي. قد يتساوى وكيلان في معدل النجاح النهائي لكن أحدهما ينهار عند أول تغيير في الواجهة، بينما يعيد الآخر التخطيط ويصل إلى الهدف بخطوات إضافية محدودة. لذلك يجب فصل «النجاح» عن «المرونة». وتقترح الورقة أن تُحقن اضطرابات منهجية أثناء التقييم: أداة غير متاحة، تغيير اسم ملف، نافذة منبثقة، انتهاء صلاحية جلسة، أو اختلاف في ترتيب عناصر GUI، ثم يقاس سلوك الاسترداد.

6. المبادئ التنفيذية لنموذج المستكشف-الملاح

6.1 رسم الخريطة أثناء المسير

لا يتعامل الوكيل مع وصف البيئة على أنه حقيقة نهائية. بعد كل ملاحظة تُحدَّث خريطة الاعتقاد: ما الأدوات المتاحة فعلاً، ما القيود، ما الحالات الموثوقة، وما المسارات التي ثبت فشلها.

6.2 سجل الرحلة كذاكرة تشغيلية

كل مسار مهم يُضغط إلى سجل منظم يحفظ المهمة والسياق والفرضيات والأفعال والملاحظات والنتيجة والدروس والثقة ونطاق الصلاحية. الهدف ليس الأرشفة بل تمكين القرار اللاحق.

6.3 ثبات الهدف ولدونة المسار

الهدف والقيود العليا تمثل البوصلة؛ أما الخطة والأدوات ومزود النموذج فهي وسائل قابلة للاستبدال. يجب أن يستطيع الوكيل تغيير الطريق دون تغيير مقصد المهمة أو تجاوز التفويض.

6.4 بعثات استطلاع صغيرة قبل الالتزام الكبير

عندما يكون عدم اليقين أو خطر عدم القابلية للعكس مرتفعاً، يفضّل الوكيل تجربة صغيرة قابلة للعكس تكشف معلومات قبل إجراء واسع.

6.5 تحويل العاصفة إلى بنية أقوى

الفشل المتكرر لا يبقى ملاحظة؛ بعد عتبة محددة يُحوّل إلى قاعدة وقائية، اختبار preflight، fallback، أو مهارة موثقة ومختبرة.

6.6 البوصلة فوق الأداة

تُقيّم الأدوات بقدرتها على خدمة الهدف ضمن الكلفة والمخاطر. لا ترتبط المعمارية بإطار أو مزود واحد.

6.7 الموانئ المؤقتة ليست الوجهة

الخطة أو الـMVP أو workflow الناجح يمثل حالة تشغيلية قابلة للمراجعة، لا بنية مقدسة. كل نجاح يمكن أن يُعاد تقييمه إذا تغيرت البيئة أو الكلفة أو المخاطر.

7. معمارية **E-NAV** ذات الطبقات السبع

تُبنى E-NAV كطبقة تحكم فوق نموذج لغوي أو منظومة متعددة النماذج. لا تشترط نموذجاً بعينه، بل تفترض وجود آلية لتوليد خطط/أفعال، منفذ أدوات، ذاكرة، ومراقب للبيئة. البنية التالية تحدد المسؤوليات التي يجب فصلها حتى يصبح سلوك «المستكشف-الملاح» قابلاً للتنفيذ والقياس.

الطبقة	الاسم الإنجليزي	المسؤولية	المخرجات الأساسية
1	Mission Compass	تثبيت الهدف، القيود، شروط النجاح، وحدود التفويض	Goal Contract، Success Tests، Safety Constraints
2	Belief Cartography Engine	تمثيل ما يُعرف وما لا يُعرف عن البيئة والأدوات	Belief Map، Uncertainty، Dependency Graph
3	Scout / Probe Layer	تنفيذ تجارب صغيرة منخفضة المخاطر لاكتساب	Probe Result، Information Gain،

الطبقة	الاسم الإنجليزي	المسؤولية	المخرجات الأساسية
		معلومات	Feasibility Evidence
4	Route Planner	إنشاء/ترتيب مسارات بديلة وتقدير الكلفة والعودة	Route Set، Checkpoints، Fallbacks
5	Operational Logbook Memory	ضغط التجربة واستخراج الدروس والقواعد القابلة للاسترجاع	Episodes، Heuristics، Failure Signatures
6	Resilience & Rerouting Engine	تشخيص الانحراف، اختيار الاسترداد، وإعادة المسار	Recovery Plan، Reroute Decision، Escalation
7	World-Effect Verification Loop	التحقق من أن الفعل غيّر العالم كما هو مقصود	Verified Effect، Postconditions، Audit Evidence

7.1 بوصلة المهمة: **Mission Compass**

تبدأ كل مهمة بعقد هدف Goal Contract يحدد النتيجة المرغوبة، ما لا يجوز تغييره، ميزانيات الزمن/الأدوات/الرموز، والأفعال التي تتطلب تصعيداً بشرياً. فصل البوصلة عن الخطة يمنع مشكلة «تكلس الخطة»؛ فالوكيل يستطيع تبديل المسار مع الاحتفاظ بشروط النجاح نفسها.

7.2 محرك الكارتوغرافيا الاعتقادية

الخريطة ليست نسخة نصية من التوثيق؛ إنها بنية حالة تحتوي على عقد وارتباطات ودرجات ثقة. مثال في بيئة حاسوب: عقد للتطبيقات والملفات والأدوات والصلاحيات، حواف للاعتماد، وخصائص مثل available=true/false، confidence، last_verified، reversibility. يمكن أن يكون جزء من الخريطة احتماليا، وجزء رمزيا، وجزء مستخرجاً من سجل الخبرة.

7.3 طبقة الاستطلاع

تولّد هذه الطبقة أفعالاً هدفها تقليل عدم اليقين: قراءة حالة، اختبار اتصال، الاستعلام عن schema، استخدام dry-run، فتح معاينة، أو تنفيذ عملية على عينة صغيرة. ويجب أن تكون هذه الطبقة محكومة بحد أقصى للكلفة والمخاطر كي لا يتحول الاستكشاف إلى تشتت.

7.4 مخطط المسار

ينتج المخطط مجموعة مسارات لا مساراً واحداً عند وجود عدم يقين مرتفع. كل مسار يحتوي checkpoints وشروط فشل وfallbacks. يمكن دمج تقنيات بحث مثل beam search أو tree search أو LATS، لكن التقييم في E-NAV لا يعتمد على جودة الاستدلال فقط، بل يضيف قابلية الاسترداد وقيمة المعلومات وخطر الأثر غير القابل للعكس.

7.5 ذاكرة سجل الرحلة

تقسم الذاكرة إلى ثلاثة مستويات: ذاكرة حدثية لمسارات خام مختصرة، ذاكرة إرشادية heuristics تستخلص قواعد عملية قابلة للاسترجاع، وذاكرة بنيوية تحتوي خرائط الأدوات والاعتمادات والحواجز الوقائية الموثقة. هذا التقسيم يحاول الجمع بين مرونة Reflexion/ERL وقابلية الحوكمة في الأنظمة الإنتاجية.

7.6 محرك الصلابة وإعادة التوجيه

عند الانحراف، لا يكرر الوكيل الفعل نفسه تلقائياً. أولاً يصنف الفشل: transient، permission، schema mismatch، unsafe/irreversible، ambiguous state، unavailable tool، أو planning error. ثم يحدد إن كان الإصلاح محلياً أو يتطلب العودة إلى checkpoint أو إعادة تخطيط كاملة أو تصعيداً بشرياً.

7.7 حلقة التحقق من الأثر

نجاح استدعاء API لا يساوي نجاح المهمة. لذلك يجب فحص postconditions: هل تغير الملف الصحيح؟ هل أُرسل البريد فعلاً؟ هل عُدلت القيمة المطلوبة فقط؟ هذه الحلقة تمنع الخلط بين «نجاح التنفيذ التقني» و«تحقق الأثر المقصود».

7.8 مخطط تدفق تشغيلي

التسلسل التشغيلي
بوصلة المهمة → خريطة الاعتقاد → تقدير عدم اليقين
↓ عند عدم اليقين المرتفع
بعثة استطلاع صغيرة → ملاحظة → تحديث الخريطة
↓
تخطيط المسار → تنفيذ → تحقق من الأثر
↓ فشل/انحراف ↓ نجاح
تشخيص + إعادة توجيه ← سجل الرحلة → ضغط الخبرة/ترقية المهارة
↓

التسلسل التشغيلي
تحديث الخريطة والذاكرة ثم الاستمرار حتى تحقق شروط النجاح

8. الصياغة الرياضية ودالة القرار

8.1 تعريف الحالة والخريطة

نفترض في الزمن t حالة بيئية خفية s_t، وملاحظة o_t، وخريطة اعتقاد B_t تمثل المعرفة التشغيلية للوكيل عن البيئة. قد تتضمن B_t توزيعاً احتمالياً، خصائص ثقة، أو رسماً بيانياً رمزياً. يختار الوكيل فعلاً a_t من مجموعة أفعال متاحة A_t، ثم يتلقى ملاحظة جديدة ويحدّث الخريطة.

$$B_{t+1} = U(B_t, a_t, o_{t+1}, L_t) \quad (1)$$

حيث U دالة تحديث تجمع بين الملاحظة الحالية وسجل الرحلة L_t. إدخال الذاكرة هنا مهم لأن معنى الملاحظة قد يعتمد على فشل مماثل سبق التحقق منه.

8.2 دالة القرار متعددة المعايير

نقترح أن يُختار الفعل أو المسار المرشح وفق منفعة مركبة لا تعتمد على التقدم المباشر نحو الهدف وحده:

$$a^*_t = \text{argmax}_a \left[\alpha G_t(a) + \beta I_t(a) + \gamma R_t(a) + \delta K_t(a) - \lambda C_t(a) - \mu H_t(a) - \nu D_t(a) \right] \quad (2)$$

الرمز	المعنى
G_t(a)	التقدم المتوقع نحو شروط النجاح Goal Progress
I_t(a)	المكسب المعلوماتي المتوقع Expected Information Gain
R_t(a)	قابلية الاسترداد Recoverability إذا فشل الفعل
K_t(a)	قيمة المعرفة القابلة لإعادة الاستخدام Reusable Knowledge Value
C_t(a)	كلفة الرموز/الزمن/الأدوات/المال
H_t(a)	المخاطر التشغيلية/الأمنية/السياسية
D_t(a)	درجة عدم القابلية للعكس أو الانحراف Irreversibility/Divergence
α…ν	أوزان تضبطها حساسية المهمة والميزانية

الميزة المقصودة من الحد I_t هي أن فعلاً لا يحقق تقدماً مباشراً قد يكون عقلانياً إذا خفض عدم اليقين بما يكفي لتجنب سلسلة أخطاء لاحقة. أما R_t وD_t فيدفعان الوكيل إلى تفضيل أفعال قابلة للعكس عندما تكون المعرفة ناقصة.

8.3 مكسب الكارتوغرافيا

$$CG_t(a) = [H(B_t) - E(H(B_{t+1}) | a)] / [C_t(a) + \varepsilon] \quad (3)$$

حيث H مقياس عدم يقين مناسب. يعبّر CG عن مقدار خفض عدم اليقين لكل وحدة كلفة. يمكن استخدام entropy في الحالات الاحتمالية، أو مقياس أبسط مثل عدد الفرضيات التشغيلية التي حُسمت بعد الاختبار.

8.4 مرونة الاسترداد

$$RE = P(\text{success} \mid \text{perturbation}) \times \exp(-\kappa \cdot \text{normalized_extra_cost}) \quad (4)$$

يقيس RE نجاح الوكيل بعد اضطراب مصطنع مع معاقبة الكلفة الإضافية. هذا أفضل من معدل النجاح وحده لأنه يميز بين وكيل يتعافى بخطوتين وآخر يعيد المهمة كاملة عشر مرات.

8.5 عائد سجل الرحلة ومعدل تكرار الفشل

$$LRY = \text{useful_retrieved_items} / \max(\text{retrieved_items}, 1) \quad (5)$$

$$RFR = \text{repeated_failure_events} / \max(\text{total_failure_events}, 1) \quad (6)$$

يرتفع LRY عندما تكون العناصر المسترجعة من الذاكرة ذات صلة فعلية بالقرار، بينما يجب أن يخفض RFR مع نضج النظام. إذا استمر نفس نمط الفشل بعد تسجيله عدة مرات فذلك يشير إلى أن الذاكرة وصفية لا تشغيلية.

8.6 حفظ الهدف

$$GPS = 1 - \text{constraint_violations} / \max(\text{decision_points}, 1) \quad (7)$$

Goal Preservation Score يقيس قدرة إعادة التخطيط على احترام عقد الهدف والقيود. قد ينجح الوكيل فنياً في الوصول إلى نتيجة لكنه يفشل في GPS إذا تجاوز شرطاً صريحاً، عدّل مورداً غير مخول، أو غيّر مخرجات جانبية غير مطلوبة.

9. الذاكرة وسجل الرحلة وتحويل الفشل إلى معرفة

9.1 لماذا لا تكفي الذاكرة الخام؟

تخزين جميع المحادثات ومسارات الأدوات ثم حقنها في السياق ليس استراتيجية تعلم مستدامة. كلما تراكمت الخبرة ارتفعت كلفة الاسترجاع وخطر جلب حالات غير مشابهة. تشير نتائج ERL إلى أهمية الاسترجاع الانتقائي، كما تظهر OEL أن المعرفة المستخلصة قد

تكون أكثر فعالية من المسارات الخام. لذلك يقترح E-NAV خط أنابيب: Episode → Diagnose → Abstract → Validate → Index → Retrieve. [15،16]

9.2 بنية سجل الرحلة

الحقل	الوصف
episode_id	معرف فريد للمسار
mission_signature	وصف مضغوط للمهمة والقيود
environment_fingerprint	إصدارات الأدوات/التطبيقات/المورد
hypotheses	الافتراضات التي بُني عليها القرار
actions	الأفعال أو الاستدعاءات الأساسية
observations	الأثر المرصود لا مجرد return code
outcome	نجاح/فشل/نجاح جزئي
failure_signature	تصنيف قابل للمطابقة مع فشل لاحق
recovery	المسار الذي أصلح المشكلة إن وجد
heuristic	قاعدة قابلة لإعادة الاستخدام
scope	حدود صلاحية القاعدة
confidence	درجة الثقة والأدلة
last_verified	آخر تحقق من استمرار صحة المعرفة

9.3 بروتوكول تحويل الفشل إلى قدرة

لا تُحوّل كل هفوة إلى مهارة جديدة، لأن ذلك يؤدي إلى تضخم معرفي. نقترح بوابة من خمس مراحل:

1. اكتشاف التكرار: يظهر failure_signature نفسه في أكثر من سياق أو تجاوز عدد مرات θ_f.
2. تفسير سببي: يصنف الوكيل السبب ويحدد ما إذا كان محلياً أو عاماً.
3. تركيب معالجة: إنشاء guardrail أو preflight check أو fallback أو skill مرشحة.
4. اختبار مضاد: تشغيل حالات موجبة وسالبة للتأكد من أن المعالجة لا تسبب ضرراً أو تمنع حالات صحيحة.

5. الترقية: تخزين العنصر في الذاكرة البنيوية مع scope وconfidence ونسخة قابلة للتراجع.

هذه الآلية تجسد مبدأ «كل عاصفة يجب أن تزيد صلابة النظام» ولكن بصورة قابلة للتدقيق: لا يكتسب النظام قدرة جديدة إلا إذا اجتازت الاختبار وأصبحت قابلة للربط بفئة واضحة من الحالات.

10. الاستكشاف المقيد وبعثات الاستطلاع

الاستكشاف المفتوح مفيد في البيئات الآمنة أو المحاكاة، لكنه قد يكون خطراً في الأنظمة الإنتاجية. لذلك تميز E-NAV بين ثلاثة أنواع من الاستطلاع: استطلاع معرفي read-only، استطلاع قابل للعكس reversible، واستطلاع يحتاج موافقة explicit approval. يحظر النظام افتراض أن «المعلومة تستحق المخاطرة» دون موازنة.

نوع الاستطلاع	مثال	الخطر	السياسة
Read-only probe	استعلام schema أو فحص حالة ملف	منخفض	يمكن تنفيذه تلقائياً ضمن الميزانية
Reversible probe	تعديل مؤقت في sandbox أو نسخة تجريبية	متوسط	يتطلب نقطة رجوع وpostcondition
Potentially irreversible	إرسال رسالة، حذف، دفع، نشر عام	مرتفع	لا يُستخدم كاستطلاع؛ يلزم تحقق/تفويض صريح

10.1 اختيار الاستطلاع

يمكن تفعيل بعثة استطلاع عندما يتجاوز عدم اليقين عتبة θ_u ويكون هناك فعل منخفض المخاطر ذي CG مرتفع. ويجب أن يوقف الاستطلاع إذا فشلت التجارب في خفض عدم اليقين بعد k محاولات، وإلا قد يدخل الوكيل في حلقة بحث لا نهائية.

$$\text{probe if } U_t > \theta_u \text{ and } \max_a CG_t(a) > \theta_g \text{ and } H_t(a) < \theta_h \quad (8)$$

10.2 الاستكشاف مقابل الاستغلال

يختلف المقترح عن ε-greedy العشوائي؛ فالاستطلاع هنا موجّه نحو فرضية تشغيلية محددة. مثال: قبل الاعتماد على أداة إرسال بريد، قد يجري الوكيل استعلاماً عن المسودات أو metadata للتأكد من الحساب والمجلد والهوية المرسلة. هذا الفعل لا يختار عشوائياً؛ بل يقلل عدم يقين له تأثير مباشر على مسار المهمة.

11. إعادة التخطيط والصلابة تحت الاضطراب

11.1 تصنيف الاضطراب

الفئة	مثال	الاستجابة المفضلة
Transient	مهلة شبكة مؤقتة	Retry محدود مع backoff
Permission	صلاحية مفقودة	تغيير المسار أو طلب تفويض؛ لا تكرار أعمى
Schema drift	تغيير API/واجهة	استطلاع schema وتحديث الخريطة
Tool unavailable	مزود أو أداة غير متاحة	اختيار بديل وفق Mission Compass
State ambiguity	الوكيل غير متأكد من الحالة الحالية	Observation/verification قبل أي فعل جديد
Planning error	تفكيك المهمة ناقص	عودة إلى checkpoint وإعادة decomposition
Safety/irreversibility	فعل قد يسبب أثراً دائماً	توقف/تصعيد/تأكيد إضافي

11.2 مستويات إعادة التوجيه

تقترح E-NAV إعادة توجيه هرمية: المستوى 0 إعادة المحاولة الآمنة؛ المستوى 1 تغيير معلمة أو أداة؛ المستوى 2 العودة إلى checkpoint؛ المستوى 3 إعادة تخطيط المسار؛ المستوى 4 إعادة تفسير الخريطة أو الهدف الفرعي؛ المستوى 5 تصعيد بشري. الهدف هو إصلاح أقل مستوى ممكن لتقليل الكلفة، مع منع retry loops.

11.3 مؤشرات الانهيار المبكر

ينبغي لمحرك الصلابة مراقبة: تكرار نفس الخطأ، زيادة عدد الخطوات دون تقدم، تناقض الملاحظات مع الخريطة، تزايد كلفة الأدوات، انخفاض ثقة الحالة، أو تجاوز checkpoint دون تحقق. هذه إشارات تدفع إلى إعادة التخطيط قبل استنزاف الميزانية.

12. التحقق من الأثر والحوكمة والسلامة

الوكيل الذي ينفذ أدوات حقيقية يحتاج إلى حوكمة تفوق مجرد تصفية النص. E-NAV يضع الأمان داخل بنية القرار نفسها عبر H_t وD_t، ويشترط postconditions بعد الأفعال المهمة. ويجب التفريق بين ثلاثة مستويات: نجاح الاستدعاء، صحة الحالة الجديدة، وتحقيق الهدف الجزئي. قد يعيد API رمز 200 لكن يكتب في مورد خاطئ؛ وقد ينجح click لكنه يفتح نافذة غير المقصودة.

12.1 قاعدة «لا تجعل الاستطلاع أثراً دائماً»

الأفعال ذات الأثر الخارجي غير القابل للعكس—مثل الإرسال، النشر، الحذف النهائي، تحويل الأموال، تغيير صلاحيات واسعة—لا تستخدم كاختبارات لاستكشاف البيئة. إذا كانت المعرفة ناقصة، ينبغي البحث عن read-only أو sandbox أو preview أو dry-run، أو التصعيد.

12.2 أدلة التحقق

بعد كل أثر مهم، يسجل الوكيل evidence: معرف مورد، حالة قبل/بعد، قيمة متوقعة/مرصودة، timestamp، والاختبار الذي أكد النجاح. هذا يجعل سجل الرحلة قابلاً للتدقيق ويمنع الذاكرة من تسجيل نجاح غير مثبت.

12.3 حدود الاستقلال

المستكشف الجيد لا يعني وكيلاً بلا حدود. التفويض جزء من الخريطة؛ فإذا لم يكن واضحاً أن فعلاً مسموح، فإن عدم اليقين نفسه سبب للتوقف أو طلب الإذن. بهذا تصبح المرونة مقيدة دستورياً بعقد المهمة، لا قدرة على تجاوز الضوابط عند تعطل المسار الأول.

13. الخوارزمية التشغيلية المقترحة

الخوارزمية التالية وصف مرجعي ويمكن تنفيذها فوق أي إطار وكلاء. الهدف منها إظهار نقاط التحكم التي تميز E-NAV عن حلقة ReAct بسيطة.

Algorithm 1 - E-NAV Control Loop

المدخلات: Mission Contract M، ميزانية B، ذاكرة L، أدوات T

1) ابنِ/استرجع خريطة الاعتقاد B_t وحدد عدم اليقين الحرج.

2) إذا كان عدم اليقين الحرج مرتفعاً: ولّد probes آمنة، قيّم CG والمخاطر، ونفّذ الأفضل.

3) حدّث B_t من الملاحظات وسجل evidence.

4) ولّد عدة مسارات عند الحاجة، مع checkpoints وfallbacks.

5) قيّم كل مسار بدالة القرار متعددة المعايير.

6) نفّذ الخطوة التالية ثم تحقق من الأثر الفعلي.

7) عند الفشل: صنف failure_signature وحدد مستوى إعادة التوجيه.

8) استرجع خبرة ذات صلة فقط؛ امنع التكرار الأعمى.

9) حدّث سجل الرحلة، واختبر إن كان نمط الفشل يستحق guardrail/skill.

10) كرر حتى تحقق شروط النجاح أو استنفاد الميزانية أو ضرورة التصعيد.

11) اختم بتقرير أثر: ما تحقق، ما لم يتحقق، وما الذي تعلمته الخريطة.

13.1 سياسة منع الدوران

يجب أن يفرض النظام حد retry لكل failure_signature، وحداً لعدد الاستطلاعات المتشابهة، ومقياساً للتقدم. إذا لم يتحسن Goal Progress أو لم يخفض uncertainty خلال نافذة n خطوات، ينتقل الوكيل إلى إعادة تخطيط أعلى مستوى أو تصعيد.

13.2 سياسة ضغط السياق

لا يُحقن سجل الرحلة بالكامل. تُسترجع عناصر وفق تشابه المهمة، تشابه البيئة، confidence، freshness، ودرجة السببية للفشل. يمكن أن تحجز بنود السلامة والحواجز الوقائية مساراً ذا أولوية أعلى من heuristics الأداء، حتى لا يسقط قيد مهم بسبب ضيق السياق.

14. بروتوكول التقييم والتجارب

لتحويل E-NAV من اقتراح هندسي إلى مساهمة تجريبية، تقترح الورقة تقييماً متعدد البيئات مع ميزانيات مضبوطة. يجب مقارنة E-NAV بخطوط أساس تستخدم النموذج نفسه والأدوات نفسها قدر الإمكان، كي لا تُنسب مكاسب النموذج الأقوى إلى المعمارية.

14.1 بيئات الاختبار

البيئة	السبب
AgentBench [7]	تنوع بيئات التفاعل وصنع القرار
WebArena [9]	مهام ويب واقعية تتطلب التنقل بين مواقع ووظائف
OSWorld [10]	تفاعل GUI حقيقي متعدد التطبيقات وأنظمة التشغيل
OSWorld-MCP [14]	مقارنة GUI مع استدعاء الأدوات/MCP
WindowsWorld [17]	مهام مهنية متعددة التطبيقات وأهداف فرعية طويلة

14.2 خطوط الأساس

- ReAct فقط.
- ReAct + Reflexion.
- ReAct + ذاكرة استرجاعية بسيطة دون Cartography.
- Planner-Executor تقليدي مع retry/fallback ثابت.
- E-NAV كامل.

14.3 اضطرابات مضبوطة

- إخفاء أداة كانت متاحة في بداية المهمة.

• تغيير اسم/موضع عنصر GUI دون تغيير الوظيفة.

• إرجاع خطأ permission بعد تقدم جزئي.

• تغيير schema أو field name في أداة.

• إدخال نافذة منبثقة أو حالة session expired.

• إجبار مسار أولي على نقطة مسدودة مع بقاء مسار بديل صالح.

14.4 المقاييس

المقياس	الاتجاه المطلوب	لماذا
Task Success	↑	النتيجة النهائية
Steps / Tool Calls	↓	الكفاءة
Token/Cost	↓	الجدوى التشغيلية
Recovery Elasticity RE	↑	التعافي من الاضطراب
Cartographic Gain CG	↑	قيمة الاستطلاع
Repeat Failure Rate RFR	↓	هل يتعلم النظام من أخطائه؟
Logbook Reuse Yield LRY	↑	جودة الاسترجاع
Goal Preservation Score GPS	↑	منع نجاح على حساب القيود
Irreversible Error Count	↓	السلامة العملية

14.5 الضبط الإحصائي

يُنصح باستخدام عدة بذور/إعادات لكل مهمة، والإبلاغ عن المتوسط وفواصل الثقة، واختبارات مقترنة عندما تكون نفس المهام مستخدمة بين المعماريات. كما يجب تثبيت ميزانية القصوى للرموز والأدوات أو الإبلاغ عن منحنيات الأداء مقابل الكلفة، لأن أي معمارية قد تبدو أفضل ببساطة إذا سُمح لها بمزيد من المحاولات.

15. دراسات الإزالة والفرضيات القابلة للتكذيب

15.1 فرضيات البحث

الفرضية	الصياغة
H1	إضافة Belief Cartography إلى ReAct تقلل الفشل الناتج عن افتراضات بيئية خاطئة مقارنة بذاكرة نصية فقط.
H2	إضافة Probe Layer ترفع النجاح تحت schema/tool uncertainty عندما تكون الأفعال الاستطلاعية منخفضة المخاطر، لكنها قد تزيد الكلفة في البيئات المستقرة.
H3	Operational Logbook مع استرجاع انتقائي يخفض Repeat Failure Rate مقارنة بإعادة حقن المسارات الخام.
H4	Resilience & Rerouting يرفع Recovery Elasticity تحت اضطرابات مضبوطة حتى إذا لم يتغير معدل النجاح على المهام السهلة.
H5	استخدام D_t وH_t يخفض الأخطاء غير القابلة للعكس، وقد يفرض مقايضة طفيفة مع السرعة.
H6	ترقية failure patterns إلى guardrails/skills بعد تحقق مضاد تحسن الأداء عبر المهام أكثر من حفظ التأملات غير المختبرة.
H7	E-NAV يحقق ميزة أكبر كلما زاد طول الأفق وتعدد التطبيقات وعدم اليقين، بينما قد لا يستحق overhead في المهام القصيرة الحتمية.

15.2 دراسات الإزالة

- E-NAV بدون Information Gain: قياس قيمة الاستكشاف الصريح.
- E-NAV بدون Logbook: اختبار ما إذا كان التحسن لحظياً فقط.
- E-NAV بدون Resilience Engine: استبداله بـ retry ثابت.
- E-NAV بدون World-Effect Verification: قياس false-success.
- E-NAV بدون Irreversibility penalty: قياس أخطاء الأثر الدائم.

• E-NAV مع ذاكرة خام بدلاً من heuristics: قياس أثر الضغط والتجريد.

15.3 معيار التكذيب

يجب رفض الادعاء الأقوى للورقة إذا لم يحقق E-NAV تحسناً ذا دلالة في RE/RFR/GPS على المهام المضطربة بعد ضبط الكلفة، أو إذا كانت المكاسب تختفي عند مساواة عدد tool calls والرموز. كما يجب اعتبار Probe Layer غير مبررة في مجال معين إذا زادت الكلفة دون خفض uncertainty أو رفع النجاح.

16. تطبيقات هندسية وريادية وبحثية

16.1 وكلاء هندسة البرمجيات

في إصلاح مستودع برمجي، الخريطة تشمل بنية المشروع، الاختبارات، الاعتماديات، وأثر التعديل. بدلاً من تحرير عدة ملفات ثم تشغيل الاختبارات متأخراً، ينفذ الوكيل استطلاعات مثل قراءة الاختبارات المتصلة، reproduction minimal، أو فحص call graph. سجل الرحلة يحتفظ بأنماط الفشل الخاصة بالمستودع، بينما يتأكد World-Effect Loop من مرور الاختبارات وعدم إفساد سلوك جانبي. تتقاطع الفكرة مع SWE-agent الذي يؤكد أهمية واجهات مناسبة للوكلاء في مهام هندسة البرمجيات. [13]

16.2 وكلاء الحاسوب وعمليات المكاتب

في مهام GUI، قد يتغير موضع الأزرار أو حالة التطبيق. هنا تصبح الكارتوغرافيا حالة مرئية متجددة؛ بعد كل خطوة رئيسية يجب إعادة رصد الحالة قبل متابعة سلسلة طويلة من النقرات. OSWorld وWindowsWorld مناسبان لاختبار هذا النوع من الاسترداد. [10،17]

16.3 الأتمتة متعددة **APIs**

في خط بيانات أو سير عمل B2B، قد تعتمد خطوة على CRM وأخرى على البريد وثالثة على التخزين. إذا تعطل مزود واحد، يحافظ Mission Compass على الهدف بينما يبحث Route Planner عن مزود أو قناة بديلة ضمن نفس التفويض. يسجل Logbook قيود rate limits وschema وfallbacks التي ثبت نجاحها.

16.4 ريادة الأعمال واختبار السوق

يمكن تطبيق المنطق نفسه خارج البرمجيات: السوق خريطة ناقصة، والحملة الصغيرة Probe، والنتائج ملاحظات، والعرض التجاري مسار قابل للتعديل. لكن المقصود هنا نموذج قرار تجريبي لا تشبيه شعري: يجب تعريف فرضية، كلفة اختبار، مقياس معلومات، شرط توقف، ثم تحديث نموذج السوق بناءً على الأدلة.

16.5 البحث العلمي

في البحث، تمثل الخريطة شبكة الفرضيات والأدلة والتناقضات. يمكن أن يعمل Scout Agent على تجارب/تحقق محدود، بينما يُحدّث Cartography Engine درجات الثقة. ويجب أن يمنع Mission Compass تحول الاستكشاف إلى تأكيد انتقائي؛ فالمسارات التي قد تكذب النظرية يجب أن تحصل على قيمة معلوماتية مرتفعة.

16.6 الأنظمة الإبداعية

في الإنتاج الفني أو تصميم الهوية، لا يكون الهدف دائماً تحسين دالة عددية واحدة. يمكن تعريف البوصلة كمجموعة قيود أسلوبية واحتياجات جمهور، والاستطلاع كنماذج أولية A/B، وسجل الرحلة كمكتبة قرارات بصرية/صوتية مع أسباب قبولها أو رفضها. الفائدة هنا الحفاظ على الاتساق مع السماح بالتجريب.

17. المناقشة والحدود

17.1 ليست كل مهمة بحاجة إلى مستكشف

المعمارية تضيف overhead: تقدير probes ،uncertainty، ذاكرة، وإعادة تخطيط. في مهمة حتمية قصيرة ذات API ثابت قد يكون مسار مباشر أكثر كفاءة. لذلك تقترح E-NAV تفعيل الاستكشاف تكيفياً عندما يبرره عدم اليقين أو خطر عدم القابلية للعكس، لا في كل خطوة.

17.2 صعوبة تقدير عدم اليقين في **LLMs**

قد تكون ثقة النموذج اللفظية غير معايرة. لذلك لا ينبغي بناء U_t على self-report فقط. يمكن دمج أدلة خارجية: تناقض نتائج الأدوات، حداثة التحقق، تعدد المرشحين، نجاح/فشل probes، أو اختلاف بين نماذج. هذه نقطة بحث مفتوحة لأنها تؤثر مباشرة في جودة الكارتوغرافيا.

17.3 خطر الذاكرة المتقادمة

المعرفة التشغيلية تتقادم بسرعة. heuristic صحيحة لنسخة أداة قد تصبح خاطئة بعد تحديث. لهذا يتطلب السجل environment_fingerprint وlast_verified وtime decay، مع إمكانية إعادة التحقق قبل تطبيق قاعدة عالية الأثر.

17.4 تحويل الفشل إلى قاعدة قد يفرط في التعميم

إذا استخلص الوكيل قاعدة من عدد قليل من الحالات، قد يمنع مسارات صحيحة لاحقاً. لذلك تشترط الورقة scope واضحاً واختباراً مضاداً. ويمكن قياس false-positive guardrails ضمن التقييم.

17.5 الاستكشاف قد يصبح مخاطرة

في المجالات الحساسة، قد تكون أبسط «تجربة» غير مقبولة. لذلك لا يدعي E-NAV أن القيمة المعلوماتية تبرر أي فعل. يتم طرح H_t وD_t كقيود/عقوبات، ويمكن جعل بعض الفئات hard constraints لا يجوز تجاوزها مهما ارتفع I_t.

17.6 الاعتماد على جودة النموذج الأساسي

كما تشير أعمال التأمل واستخدام الأدوات، فإن المعمارية لا تعوض بالكامل عن نموذج أساسي ضعيف في فهم التعليمات أو grounding. وظيفة E-NAV هي إدارة القرار والذاكرة والمرونة، لا خلق قدرة غير موجودة من الصفر. [4،15]

18. برنامج البحث المستقبلي

- تنفيذ مرجع مفتوح المصدر لـ E-NAV فوق بيئة أدوات موحدة مع tracing كامل.
- تعلم أوزان دالة القرار $\alpha...\nu$ من بيانات المهام بدلاً من ضبطها يدوياً.
- استخدام uncertainty calibration خارجية أو ensembles لتفعيل الاستطلاع بصورة أكثر دقة.
- بناء خرائط اعتقاد هجينة: رسم رمزي + embeddings + probabilities + timestamps.
- تعلم حدود صلاحية heuristics تلقائياً وتقدير خطر التعميم خارج المجال.
- اختبار نقل سجل الرحلة بين نماذج مختلفة: هل المعرفة التشغيلية model-agnostic فعلاً؟
- دراسة multi-agent cartography حيث يرسل عدة scouts أدلة متنافسة ويقوم navigator بدمجها.
- دمج active inference أو Bayesian experimental design بصورة أكثر رسمية مع القيود التشغيلية.
- بناء benchmark خاص بالاضطرابات والتعافي يقيس route plasticity بشكل مباشر.
- اختبار ترسيخ الخبرة في الأوزان على غرار OEL مقابل إبقائها في ذاكرة خارجية، وقياس مقايضة المرونة/الكلفة/الخصوصية.

18.1 سؤال أعمق: هل الذكاء خريطة أم قدرة على صنع الخرائط؟

تفتح المعمارية سؤالاً نظرياً: في البيئات المفتوحة قد تكون قيمة النظام العام أقل ارتباطاً بحجم «ما يعرفه مسبقاً» وأكثر ارتباطاً بقدرته على تحويل عدم اليقين إلى تجارب آمنة، ثم تحويل التجارب إلى معرفة متراكمة. بهذا المعنى، يمكن تعريف جانب من الذكاء العملي بوصفه كفاءة مستمرة في بناء نموذج قابل للتصحيح للعالم تحت موارد وحدود فعلية.

19. الخاتمة

قدمت هذه الورقة صياغة هندسية لعقلية «المستكشف-الملاح» في أنظمة الذكاء الاصطناعي الوكيلية. جوهر الاقتراح هو أن الخريطة لا ينبغي أن تكون مقدمة ثابتة للعمل، بل حالة اعتقادية تتطور مع كل ملاحظة؛ وأن الفعل لا يقيم فقط بما يحققه من تقدم مباشر، بل أيضاً بما يكشفه من معلومات، وبمدى قابليته للاسترداد، وبالمعرفة التي يتركها للمستقبل.

تتكون E-NAV من بوصلة مهمة تفصل الهدف عن المسار، محرك كارتوغرافيا يحدّث المعرفة التشغيلية، طبقة استطلاع مقيدة، مخطط مسارات، سجل رحلة قابل للاسترجاع والترقية، محرك صلابة وإعادة توجيه، وحلقة تحقق من الأثر. تم اقتراح دالة قرار ومقاييس وفرضيات ودراسات إزالة تجعل الإطار قابلاً للتجربة والتكذيب بدلاً من أن يبقى تشبيهاً فلسفياً.

إذا أثبتت التجارب المقترحة أن E-NAV تخفض تكرار الفشل وترفع مرونة الاسترداد وحفظ الهدف تحت اضطرابات واقعية دون زيادة غير مقبولة في الكلفة، فستدعم النتيجة رؤية مفادها أن الجيل التالي من الوكلاء لا يحتاج فقط إلى «عقول تخطط»، بل إلى نظم ملاحة تتعلم الخريطة وهي تتحرك فيها، وتكتب سجل الرحلة، وتعود من العاصفة أكثر صلابة مما دخلتها.

الخلاصة النظرية

الوكيل المتقدم ليس النظام الذي يمتلك خريطة كاملة مسبقاً، بل النظام الذي يمتلك بروتوكولاً موثوقاً لصنع الخريطة، اختبارها، تصحيحها، وتغيير المسار دون فقدان البوصلة.

المراجع

[1] Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2022/2023). ReAct: Synergizing Reasoning and Acting in Language Models. arXiv:2210.03629.

[2] Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., & Narasimhan, K. (2023). Tree of Thoughts: Deliberate Problem Solving with Large Language Models. arXiv:2305.10601.

[3] Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language Agents with Verbal Reinforcement Learning. arXiv:2303.11366.

[4] Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative Agents: Interactive Simulacra of Human Behavior. arXiv:2304.03442.

[5] Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., & Anandkumar, A. (2023). Voyager: An Open-Ended Embodied Agent with Large Language Models. arXiv:2305.16291.

[6] Zhao, A., Huang, D., Xu, Q., Lin, M., Liu, Y.-J., & Huang, G. (2023). ExpeL: LLM Agents Are Experiential Learners. arXiv:2308.10144.

[7] Liu, X., Yu, H., Zhang, H., et al. (2023). AgentBench: Evaluating LLMs as Agents. arXiv:2308.03688.

[8] Zhou, S., Xu, F. F., Zhu, H., et al. (2023). WebArena: A Realistic Web Environment for Building Autonomous Agents. arXiv:2307.13854.

[9] Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H., & Wang, Y.-X. (2023). Language Agent Tree Search Unifies Reasoning, Acting, and Planning in Language Models. arXiv:2310.04406.

[10] Xie, T., Zhang, D., Chen, J., et al. (2024). OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments. arXiv:2404.07972.

[11] Da Costa, L., Tenka, S., Zhao, D., & Sajid, N. (2024). Active Inference as a Model of Agency. arXiv:2401.12917.

[12] de Vries, B., Nuijten, W., van de Laar, T., et al. (2025). Expected Free Energy-based Planning as Variational Inference. arXiv:2504.14898.

[13] Yang, J., Jimenez, C. E., Wettig, A., Lieret, K., Yao, S., Narasimhan, K., & Press, O. (2024). SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. arXiv:2405.15793.

[14] Jia, H., Liao, J., Zhang, X., et al. (2025). OSWorld-MCP: Benchmarking MCP Tool Invocation In Computer-Use Agents. arXiv:2510.24563.

[15] Allard, M.-A., Teinturier, A., Xing, V., & Viaud, G. (2026). Experiential Reflective Learning for Self-Improving LLM Agents. arXiv:2603.24639.

[16] Ye, T., Dong, L., Dong, Q., Wu, X., Huang, S., & Wei, F. (2026). Online Experiential Learning for Language Models. arXiv:2603.16856.

[17] Li, J., Li, Y., Zhao, C., Xu, Z., Hu, B., & Zhang, M. (2026). WindowsWorld: A Process-Centric Benchmark of Autonomous GUI Agents in Professional Cross-Application Environments. arXiv:2604.27776.

[18] Guo, Z., et al. (2025). MIRROR: Multi-agent Intra- and Inter-Reflection for Optimized Reasoning in Tool Learning. arXiv:2505.20670.

الملحق أ: تعريف المتغيرات والمقاييس

العنصر	التعريف العملي	طريقة القياس المقترحة
Belief Map B_t	المعرفة الحالية عن الأدوات والحالة والاعتماديات والثقة	Graph/structured state + confidence + last_verified
Uncertainty U_t	مقدار الغموض التشغيلي المؤثر في القرار	Entropy أو عدد فرضيات حرجة غير محسومة
Goal Progress G_t	المسافة إلى شروط النجاح	نسبة subgoals المتحققة مع weights
Information Gain I_t	خفض متوقع في عدم اليقين	Δ uncertainty بعد probe
Recoverability R_t	إمكانية العودة دون أثر دائم	تصنيف/احتمال مبني على نوع الفعل وcheckpoint
Risk H_t	خطر أمني/تشغيلي/سياساتي	Risk model أو hard constraints
Irreversibility D_t	صعوبة عكس أثر الفعل	read-only … 1 irreversible 0
CG	مكسب الكارتوغرافيا لكل كلفة	المعادلة (3)
RE	مرونة الاسترداد	المعادلة (4)
LRY	عائد استرجاع سجل الرحلة	المعادلة (5)
RFR	معدل تكرار الفشل	المعادلة (6)
GPS	حفظ الهدف والقيود	المعادلة (7)

أ.1 مثال مصغر

مهمة: تحديث تقرير في مستند ثم إرساله. الخريطة الأولية تقول إن أداة المستند وأداة البريد متاحتان، لكن هوية المرسل غير مؤكدة. إذا كان الإرسال غير قابل للعكس، فإن E-NAV لا يستخدم «إرسال رسالة تجريبية» كprobe. بدلاً من ذلك ينفذ استعلام read-only عن الحساب/المسودات، يحدث B_t، ثم يعدل المستند، يتحقق من محتواه، ينشئ مسودة، يتحقق من المستلم والمرفق، وبعد تحقق شروط التفويض ينفذ الإرسال. إذا فشلت أداة البريد بسبب session expired، يصنف الفشل ويعيد المصادقة أو يصعد بدلاً من تكرار send.

الملحق ب: نموذج سجل الرحلة

قالب بيانات مقترح

episode_id: ENAV-2026-XXXX

mission_signature: <ملخص الهدف والقيود>

<environment_fingerprint: <app/tool/model versions

critical_unknowns: <ما الذي لم يكن معروفاً>

probes: <الاختبارات المنفذة ونتائجها>

route: <المسار المختار + checkpoints>

<actions: <tool calls/actions

<verified_effects: <postconditions/evidence

outcome: success | partial | failure

<failure_signature: <taxonomy code

recovery: <ما الذي أصلح المسار>

heuristic: <قاعدة قابلة لإعادة الاستخدام>

scope: <متى تنطبق ومتى لا تنطبق>

confidence: 0..1

<last_verified: <timestamp

promotion_status: episode | heuristic | guardrail | skill

ب.1 مثال لقاعدة مُرقاة

إذا ظهر فشل permission نفسه مرتين في أداة كتابة بينما توجد واجهة read-only تعمل، يمكن تركيب guardrail: «قبل أول فعل كتابة على المورد X، تحقق من write capability عبر metadata endpoint؛ إذا كانت الصلاحية غير مؤكدة، لا تنفذ write واطلب مساراً بديلاً/تفويضاً». بعد اختبار القاعدة على حالات تملك الصلاحية وأخرى لا تملكها، تُرقى من heuristic إلى preflight guardrail مع نطاق أداة/نسخة محدد.

ملاحظة حول النزاهة العلمية

جميع الأرقام المنسوبة إلى الأعمال السابقة تعود إلى نتائج تلك الأعمال وليست نتائج لهذه الورقة. وبالأخص، الأرقام المتعلقة بـ OSWorld-MCP وWindowsWorld مأخوذة من الملخصات والتقارير الأصلية المشار إليها. أما معادلات E-NAV ومقاييس CG وRE وLRY وRFR وGPS فهي مقترحات هذه الورقة وتحتاج إلى تحقق تجريبي مستقل قبل استخدامها كدليل على التفوق.