A Wind Tunnel for Clinical AI Agents: A Proposal to Simulate the Workflow Before Deployment

Nitchakorn Tangsathapornpanich PaperOrchestra*

Draft. This is an AI-written draft generated from a proposal. It has not yet been revised by the human author and is to be refined further.

Abstract

Health systems are beginning to place AI agents beside clinical staff, while the benchmarks commonly used to judge those agents score task completion one task at a time. Such a score does not report what the agent does to the workflow around it: which human queue receives its escalations, how long a hazardous case waits there, and whether a clinician is accountable for a decision they never saw. This paper is a proposal and reports no experimental results. We argue that the unit of pre-deployment evaluation should be the workflow, and that the clinicians and administrators who do the work should be able to run that evaluation themselves. We propose a "wind tunnel": a discrete-event simulation of a human and AI workflow, driven by a workflow written as data, an agent under test whose actions are read from recorded system state, and patient behaviour specified by archetype. It is designed to report queue waits per human role, the share of decisions made without a clinician in the accountable role opening the case, and the escalation and staffing thresholds beyond which a review queue is no longer stable. A worked example with assumed inputs and a textbook queueing formula illustrates such a threshold; we list the assumptions under which the formula alone would suffice. We state eight design rules, a four-step evaluation plan, the outcomes that would count against the proposal, and what exists today: a prototype scenario set and state verifier that have not been run against any model.

1 Introduction

Large language model agents are moving from answering questions to acting inside clinical and administrative workflows. It has been argued that such agents should be assessed in simulations of clinical settings for their effect on clinical workflows, not by benchmarks built from standardised test questions [19]. A qualitative study based on interviews with developers, implementers and end users reports an evaluation blind spot that prioritises technical benchmarks over sociotechnical safety, and notes that agentic systems in health care currently operate under near-total human oversight [2].

That human oversight is a workload. Earlier clinical decision support shows how much work a system's output can create for the people who must act on it. A widely implemented proprietary sepsis model generated alerts for 18% of hospitalised patients while not identifying 67% of patients with sepsis [30]; alert fatigue limits the effectiveness of medication safety alerts [12]; and in one multispecialty practice, messages generated by the record system accounted on average for 114 of the 243 in-basket messages a physician received per week [26]. For predictive models the dependence on capacity has been studied directly: work capacity constraints can significantly reduce the net benefit of a workflow triggered by a model's predictions [14], and simulation has been used to quantify the utility gained from placing a model in a clinical workflow [31].

An agent adds two things to this picture. First, it decides when to hand a case to a human, so its escalation behaviour, together with the case mix, sets the arrival rate of a human queue. An agent that escalates every uncertain case may look cautious when scored one case at a time; where those escalations go to one reviewer, they share that reviewer's day. Second, a deployment may allow an agent to complete an action under a clinician's authority without that clinician seeing the case, and as oversight is relaxed from

*PaperOrchestra is an AI paper-writing pipeline, not a person. It drafted this text from the first author's proposal.

the near-total level reported today, this becomes the case to plan for. We call a clinician *in the loop* for a decision when they opened the case before the decision took effect, and *on the hook* when their organisation holds them accountable for it. A deployment could leave a clinician on the hook without being in the loop, and a benchmark that scores task completion alone does not report it.

Position. We hold two positions. The unit of pre-deployment evaluation for a clinical AI agent should be the workflow: the agent, the staff, the queues and the handoffs together. And the people who do the work, clinicians and administrators, should be able to change the workflow design and test the change themselves, because they know the workflow and they carry the accountability.

Proposal. We propose a wind tunnel for clinical AI agents: a simulator in which a proposed human and AI workflow is run before any patient is exposed to it. This paper reports no results; the engine is not built, and Section 6 states what exists. The component ideas come from prior work, credited in Section 2. We propose combining them in an instrument meant to be operated by clinical leaders, with accountability as a reported output. We have not surveyed prior work on measuring accountability and do not claim that this is new.

2 Related Work

Benchmarks for clinical agents. Agent benchmarks place a model in an interactive environment and score task completion. MedAgentBench provides 300 clinically derived tasks over the records of 100 patients [13]. AgentClinic evaluates models in simulated clinical environments with patient interaction and tool use [23]; PhysicianBench comprises 100 long-horizon physician tasks in an electronic health record environment [17]; PatientAgentBench evaluates patient-facing agents conversing with a simulated patient [28]; and HealthAdminBench and χ-Bench cover administrative and operational workflows [4, 5]. Outside health care, τ-bench scores an agent by comparing the database state at the end of a conversation with an annotated goal state and measures reliability over repeated trials [32]; our plan adopts both practices in Step 1. As their abstracts describe them, none of these benchmarks reports staff workload or queueing outcomes; even where handoffs between roles are part of the task [5], the score is task completion.

Usefulness simulation. A separate line of work evaluates predictive models inside their workflows. Net benefit under capacity limits has been analysed for an advance care planning workflow [14]. A simulation of patient selection showed that a provider's schedule and the timing of postoperative predictions can have major effects on cohort size and potential cost reductions [20], an approach a commentary describes as going beyond performance metrics [7]. Formulae have been given for the net benefit achievable under resource constraints [25], and APLUS offers a reusable simulation engine with a workflow specification language [31]. Our proposal is closest to APLUS. It differs in what is placed in the workflow, an agent that takes actions and decides when to involve a human in place of a model that emits a score, and in two outputs, the queue produced by the agent's escalations and the accountability trace. Extending the APLUS engine is one way to build it.

Congestion in AI and human pipelines. The interaction at the centre of our worked example is studied in prior work. Lykouris and Weng model a pipeline in which an AI classifies jobs and admits some for review by humans with limited capacity [18]. Wang and Rachev treat escalation to scarce human override capacity as a queueing control problem and identify a region of the arrival and drift parameters beyond which no policy can maintain safety standards without the queue growing without bound [29]. Related analyses cover a language-model judge ahead of human reviewers [16] and rework after AI-assisted tasks [3]. Queueing theory has been applied to the wait-time savings of radiology triage software, with a simulation tool for that workflow [27]. Separately, learned deferral between an AI model and a clinical workflow has been evaluated on clinical imaging data [8], and automation bias is a documented risk when humans review automated output [10]. The queueing analyses establish that congestion and deferral interact. Our proposal is an instrument for applying that reasoning to a specific clinic's agent workflow.

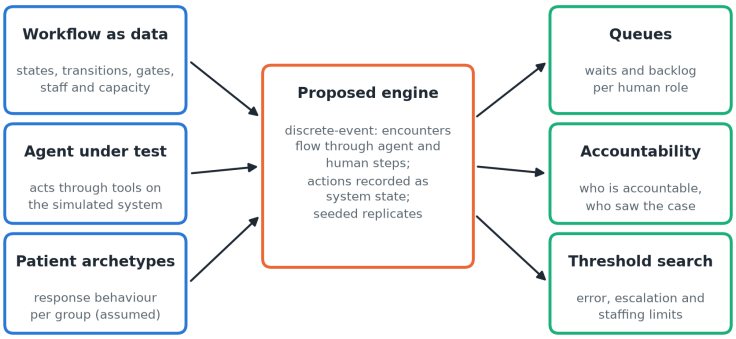


Figure 1: The proposed wind tunnel. This is a design, not a built system. Inputs (left) are a workflow written as data, an agent under test and patient archetypes. The engine (centre) would move encounters through agent and human steps and record every action as system state. Outputs (right) are queue behaviour per human role, an accountability trace and the thresholds found by search.

Language models inside simulations. Generative agent-based models couple a mechanistic model with a language model that represents human decisions [9], and simulated clinical environments populated by language-model agents have been used to improve and evaluate doctor agents [15, 1]. Grouping simulated people into archetypes has been proposed as a way to balance behavioural complexity against computational cost when language models drive agent behaviour [6]; we borrow the grouping but set each archetype's behaviour from a table of stated assumptions. A framework from another domain separates diagnosis by a language model from control by deterministic formulas so that decisions can be attributed to inspectable rules [33]. Heuristic validation of language-model simulations of people lacks the formal statistical guarantees typically required for confirmatory research [11], which is one reason our design keeps the language model out of the patient model.

3 Proposed Design

Figure 1 shows the proposed simulator. Everything in this section describes a design; none of it is built.

Inputs. The *workflow* would be written as data: the states an encounter can be in, the transitions between them, the human roles with their capacity per day, the gates at which an agent's decision is checked, the rescue path for encounters that stall, and, for each kind of decision, the role the organisation holds accountable. Writing the workflow as data follows the workflow specification language of APLUS [31]; making that file readable and editable by a clinician is our added requirement. The *agent under test* would be used in two modes. In measurement mode (Step 1 of Section 5) it runs live on scripted vignettes, connected through tools that write to the simulated system, and yields rates of clearance, escalation and missed hazard. In simulation mode those rates, not the live agent, drive each simulated encounter, so that they can also be varied across ranges. Running the live agent inside every simulated encounter is outside this plan, because the design has no generator of case content. For a reference agent we would follow, by analogy, the separation of diagnosis from control [33]: the language model classifies the case and fixed protocol rules choose the action. That no conversation can override a safety gate is our own requirement. *Patient archetypes* would specify how groups of patients respond to outreach, for example how often and how quickly they reply, from a table of stated assumptions.

Engine. A discrete-event core would move each encounter through agent and human steps, with a queue in front of each human role. Runs would be seeded and repeated. An agent's action would count only when it appears in the recorded system state, not on the agent's own account of what it did. Tool errors, abandoned encounters and tasks still unresolved when the simulated period ends would stay in the denominator of every rate.

Table 1: Design rules for the proposed simulator. These are commitments made before building, not findings.

Rule	Aim
The workflow is a data file a clinician can read and edit	Let the people who do the work change the design under test
For the reference agent, the model classifies and fixed rules act; gates cannot be overridden by conversation	Keep actions auditable, and keep a persuasive message from bypassing a safety check
Verified actions are read from system state	Avoid relying on an agent's report of its own action
Patient behaviour is set per archetype from stated assumptions	Keep unvalidated language-model behaviour out of the patient model
Failures stay in the denominator	Avoid success rates inflated by dropped encounters
Time is modelled at the resolution of the claim	A weekly model cannot support a claim about hours
Results are reported across ranges of assumptions, not only across seeds	Show uncertainty that comes from assumed inputs
Output is decision support, never an outcome forecast	Compare designs; make no prediction of patient outcomes

Outputs. The first output would be queue behaviour per human role: waits, backlog, and the wait of hazardous cases specifically. The second would be an accountability trace. A *decision* is an action that changes a patient's care, such as advancing a dose. For each decision the trace would record the accountable role declared in the workflow file and whether a person in that role opened the case before the action took effect. The reported measure is the share of all decisions in the run taken without that. Placing a case in a review queue is not counted as a decision. Where every gate is fixed, this share follows from the workflow design and the agent's measured clearance rate, with no simulation needed; a simulation is needed where presentation depends on run-time behaviour, for example when a case is released after waiting longer than a set limit. We present it as one side of a trade-off: routing more decisions past a clinician lowers the share and raises that role's queue. The third output would be a threshold search: across ranges of agent error rate, escalation rate and staffing, the simulator would locate the boundaries beyond which a queue stops being stable or a stated time-to-review target is missed.

Design rules. Table 1 lists eight rules the design commits to. They are constraints set in advance from the literature above and our own judgement. None has been tested.

4 Worked Example: A Threshold That Depends on Staffing as Well as the Agent

This section is an illustration. Every input is an assumed value, the calculation is a textbook formula, and nothing here is a measurement or a simulator output.

Consider an agent that handles n patient encounters per working day and escalates a fraction e of them to a pool of c human reviewers, each able to complete μ reviews per day. In the long run the review queue is stable only if

$$n\,e < c\,\mu. \tag{1}$$

Below this limit, if escalations arrive at random, review times are exponentially distributed, no case leaves the queue unreviewed and unfinished reviews carry over to the next day, the steady-state mean wait before review begins follows from the Erlang C formula for a queue with c servers [24]. At or above the limit the backlog grows for as long as the conditions hold.

Assume $n = 200$ encounters per day, 12 minutes per review and 6 hours of review time per reviewer per day, so $\mu = 30$; and assume, for simplicity, that escalations arrive only during review hours and evenly across them. Waits are then in minutes of reviewer working time, not clock time. Equation (1) gives a limit of $e = 0.15$ for one reviewer, 0.30 for two and 0.45 for three. Figure 2 shows the steady-state mean wait below each limit; the curves leave the plotted range near the limit. With one reviewer the mean wait is 6.0 minutes

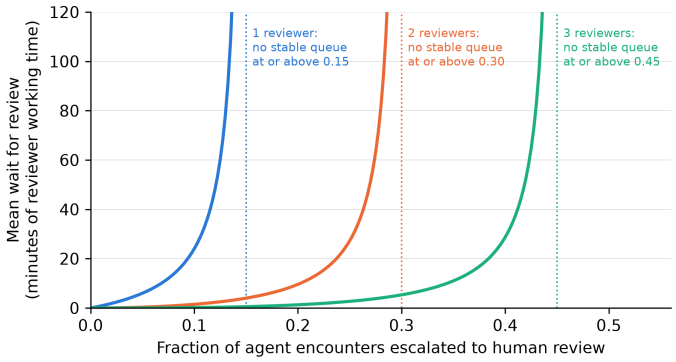


Figure 2: Worked example with assumed inputs: 200 agent encounters per day, 12 minutes per review, 6 hours of review time per reviewer per day. Steady-state mean wait before review begins, from the Erlang C formula, by the fraction of encounters the agent escalates, for one, two and three reviewers. Dotted lines mark the stability limit of Eq. (1). This is an illustration, not a result.

at $e = 0.05$, 24.0 minutes at $e = 0.10$ and 168.0 minutes at $e = 0.14$. With two reviewers it is 9.6 minutes at $e = 0.20$ and 171.1 minutes at $e = 0.29$.

Two points follow from the queueing model. First, the wait rises slowly and then steeply: under these assumptions, moving the escalation fraction from 0.10 to 0.14 with one reviewer multiplies the mean wait before review begins by seven, from 24.0 to 168.0 minutes. Second, the quantity that matters at the queue is the total escalation fraction, whatever share of it is hazardous. If hazardous and benign escalations share one queue, take equally long to review and are reviewed first come, first served (three further assumptions), a hazardous case waits as long as any other, so needless escalations lengthen the wait of the cases the review exists to catch. A priority lane changes this, and the agent's priority assignment then becomes an output to test. A third point is our conjecture and does not follow from the model: a benchmark that scores each case alone and does not penalise needless escalation would reward this caution. We have not examined whether current benchmarks do. Prior work analyses the interaction of deferral and congestion formally [18, 29]; the points above are our reading of the simple model, not results of those papers.

The example also marks where a formula stops. The threshold above needs only a per-case escalation fraction and three staffing numbers, and where its assumptions hold no simulator is needed. We expect a real workflow to depart from them in ways that may matter: arrivals may cluster after clinic sessions and laboratory runs, reviewers may be interrupted, urgent cases may be given priority, patients may fail to respond, and an agent's escalation rate may vary with the case mix. The example sets no time-to-review target, so it does not show that any point below the limit is unsafe; a target is an input a clinical lead must supply. Whether these departures change the answer enough to justify a simulation is what the plan below is meant to find out.

5 Evaluation Plan

The plan has four steps and is sized for a five-week build.

Step 1: measure agent behaviour. We will measure the escalation behaviour of two or three current models on clinician-labelled vignettes for one medication-titration follow-up workflow, with repeated runs per vignette, scoring each episode from recorded system state. The measured quantities are the rates of correct clearance, correct escalation, false escalation and missed hazard, with intervals. A second clinician will label a subset so that agreement between labellers can be reported.

Step 2: verify the engine. We will check the engine against Eq. (1) and the Erlang C wait in the simple case of Section 4, where the answer is known. This is model verification in Sargent's terms; conceptual

model validity will be reviewed with clinicians who run the workflow, and operational validity is outside this plan because it needs clinic data we do not have [22]. The simulation will be reported against the STRESS (Strengthening the Reporting of Empirical Simulation Studies) checklist [21].

Step 3: locate measured agents, and compare simulator with formula. We will map the boundary at which the shared reviewer queue stops being stable as a function of escalation rate and staffing. A model's overall escalation rate depends on its measured rates and on the share of encounters that are hazardous, which the vignettes do not tell us. We will therefore place each measured model on the map as a band across a stated range of that share, and report the result whichever side the band falls on. We will then add the departures listed in Section 4 one at a time, each over a stated range, and report how far the simulated boundary and waits move from the formula's.

Step 4: sessions with the intended users. We will hold working sessions with 5 to 10 clinicians and health-system leaders, who will change agent rules and staffing in the simulator for the medication-titration workflow, set to the staffing and volumes of their own setting. The task is fixed in advance: each participant changes one staffing number and one agent rule in the workflow file and states where the queue builds and which decisions are taken without the accountable role seeing the case. We will record who completes it without help, and which of their questions the simulator could not answer.

What would count against the proposal. Three outcomes would weaken it, and we will report each if it occurs. If, across the ranges in Step 3, the simulator's boundary and waits stay close to the formula's, the formula is enough for this workflow and the engine adds little. If the measured models' escalation rates put the queue well inside the stable region for every staffing level a clinical lead considers realistic, congestion is not the binding problem for this workflow. If most participants in Step 4 cannot complete the fixed task without an engineer, the claim that clinicians can operate the instrument fails. The tolerance for "close", the staffing levels and the range for each departure and for the hazardous share will be fixed with the clinical lead, with the source of each range recorded, before any model is measured in Step 1. The plan does not test the accountability measure beyond computing it.

6 Limitations and Current Status

Status. No experiment has been run and the engine is not built. What exists is a prototype scenario file of 8 vignettes in 4 archetypes for the medication-titration workflow, of which 3 are labelled as correctly cleared without review and 5 as correctly escalated to a pharmacist; a prototype sandbox with 3 tools (ask one clarifying question, clear and advance, escalate to the pharmacist queue); and a verifier that scores an episode from the state those tools wrote. None has been run against a model, and no second clinician has labelled the vignettes. Three exploratory browser simulations of one clinic morning exist; they are not validated and we take no results from them.

Limitations. A simulator's output is only as good as its inputs. Patient response behaviour, review times and arrival patterns will at first be assumptions, which is why the design calls for results across ranges of assumptions. Operational validation needs timing and queue data from a real clinic. Eight vignettes are too few to estimate an escalation rate, and vignettes written by clinicians may not match the distribution of real encounters. The accountability trace records whether a case was opened, which is weaker than whether it was attended to, and the design models reviewers as working at a fixed rate; automation bias [10] and review quality under load are outside it. Related work on responsibility for automated decisions and on facilitated simulation with clinicians is not covered here. Finally, a five-week build covers one workflow, and nothing here shows that the approach carries to a second.

7 Conclusion

We have argued that a clinical AI agent judged safe one case at a time could still overload the people who review its escalations, and could leave a clinician accountable for decisions they did not see, and that a task-completion score does not report either effect. We have proposed a simulator intended to evaluate the workflow as a whole and to be operated by the people who run it, stated its design rules, and set out how we will test it and what would count against it. The proposal is unevaluated. The next step is Step 1 of the plan: measured agent behaviour on labelled vignettes.

References

[1] Mohammad Almansoori, Komal Kumar, and Hisham Cholakkal. MedAgentSim: Self-evolving Multi-agent Simulations for Realistic Clinical Interactions. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2025*, Lecture Notes in Computer Science, pages 362–372. Springer Nature Switzerland, 2025. arXiv:2503.22678.

[2] Gabriela Aránguiz-Dias, K. Meimandi, Allie Griffith, Carolina Aránguiz Dias, Grace Ra Kim, Lana Saadeddin, and Mykel J. Kochenderfer. The Doctor Will (Still) See You Now: On the Structural Limits of Agentic AI in Healthcare. *Conference on Fairness, Accountability and Transparency*, 2026.

[3] Silvia Bartolucci and P. Vivo. Queue & AI: When Faster Tasks Slow Down the Workflow. *arXiv.org*, 2026.

[4] Suhana Bedi, Ryan Welch, Ethan Steinberg, Michael Wornow, Taeil Matthew Kim, Haroun Ahmed, Peter Sterling, Bravim Purohit, Qurat Akram, Angelic Acosta, Esther Nubla, Pritika Sharma, Michael A. Pfeffer, Sanmi Koyejo, and Nigam H. Shah. HealthAdminBench: Evaluating Computer-Use Agents on Healthcare Administration Tasks. *arXiv preprint arXiv:2604.09937*, 2026.

[5] Haolin Chen, Deon Metelski, Leon Qi, Tao Xia, Joonyul Lee, Steve Brown, Kevin Riley, Frank Wang, T. Y. Alvin Liu, Hank Capps, et al. χ-Bench: Can AI Agents Automate End-to-End, Long-Horizon, Policy-Rich Healthcare Workflows? *arXiv preprint arXiv:2605.16679*, 2026.

[6] Ayush Chopra, Shashank Kumar, Nurullah Giray Kuru, Ramesh Raskar, and Arnau Quera-Bofarull. On the Limits of Agency in Agent-based Models. In *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 500–509, 2025. arXiv:2409.10568.

[7] James A. Diao, L. Wedlund, and J. Kvedar. Beyond performance metrics: modeling outcomes and cost for clinical machine learning. *npj Digital Medicine*, 2021.

[8] K. Dvijotham, J. Winkens, Melih Barsbey, Sumedh Ghaisas, Robert Stanforth, Nick Pawlowski, Patricia Strachan, Zahra Ahmed, Shekoofeh Azizi, Yoram Bachrach, Laura Culp, M. Daswani, J. Freyberg, Christopher J. Kelly, A. Kiraly, Timo Kohlberger, S. McKinney, Basil Mustafa, Vivek Natarajan, K. Geras, J. Witowski, Z. Qin, J. Creswell, S. Shetty, Marcin Sieniek, Terry Spitz, G. Corrado, P. Kohli, Taylan Cemgil, and A. Karthikesalingam. Enhancing the reliability and accuracy of AI-enabled diagnosis via complementarity-driven deferral to clinicians. *Nature Medicine*, 2023.

[9] Navid Ghaffarzadegan, Aritra Majumdar, Ross Williams, and Niyousha Hosseinichimeh. Generative agent-based modeling: an introduction and tutorial. *System Dynamics Review*, 40(1):e1761, 2024. arXiv:2309.11456.

[10] Kate Goddard, A. Roudsari, and J. Wyatt. Automation bias: a systematic review of frequency, effect mediators, and mitigators. *J. Am. Medical Informatics Assoc.*, 2012.

[11] Jessica Hullman, David Broska, Huaman Sun, and Aaron Shaw. This human study did not involve human subjects: Validating LLM simulations as behavioral evidence. *arXiv preprint arXiv:2602.15785*, 2026.

[12] Mustafa I. Hussain, T. Reynolds, and Kai Zheng. Medication safety alert fatigue may be reduced via interaction design and clinical role tailoring: a systematic review. *J. Am. Medical Informatics Assoc.*, 2019.

[13] Yi-Xing Jiang, K. Black, Gloria Geng, Danny Park, James Zou, Andrew Y. Ng, and Jonathan H. Chen. MedAgentBench: A Realistic Virtual EHR Environment to Benchmark Medical LLM Agents, 2025. arXiv preprint arXiv:2501.14654.

[14] Kenneth Jung, Sehj Kashyap, A. Avati, S. Harman, Heather Shaw, Ron C. Li, Margaret Smith, K. Shum, J. Javitz, Yohan Vetteth, Tina Seto, S. Bagley, and N. Shah. A framework for making predictive models useful in practice. *Journal of the American Medical Informatics Association*, 2020.

[15] Junkai Li, Yunghwei Lai, Weitao Li, Jingyi Ren, Meng Zhang, Xinhui Kang, Siyu Wang, Peng Li, Ya-Qin Zhang, Weizhi Ma, and Yang Liu. Agent Hospital: A Simulacrum of Hospital with Evolvable Medical Agents. *arXiv preprint arXiv:2405.02957*, 2024.

[16] Rui-Han Lin and Ji-Heng Zhang. When to Screen, When to Bypass: LLM-Judges in Resource-Scarce AI-Human Workflow, 2026. arXiv preprint arXiv:2603.13870.

[17] Ruoqi Liu, Imran Q. Mohiuddin, Austin J. Schoeffler, Kavita Renduchintala, Ashwin Nayak, Prasantha L. Vemu, Shivam C. Vedak, Kameron C. Black, John L. Havlik, Isaac Ogunmola, Stephen P. Ma, Roopa Dhatt, and Jonathan H. Chen. PhysicianBench: Evaluating LLM Agents in Real-World EHR Environments. *arXiv preprint arXiv:2605.02240*, 2026.

[18] Thodoris Lykouris and Wen-Tao Weng. Learning to Defer in Congested Systems: The AI-Human Interplay, 2024. arXiv preprint arXiv:2402.12237.

[19] Nikita Mehandru, B. Miao, E. Rodriguez Almaraz, Madhumita Sushil, A. Butte, and A. Alaa. Evaluating large language models as agents in the clinic. *npj Digital Medicine*, 2024.

[20] V. Mišić, Kumar Rajaram, and E. Gabel. A simulation-based evaluation of machine learning models for clinical decision support: application and analysis using hospital readmission. *npj Digital Medicine*, 2021.

[21] T. Monks, C. Currie, B. Onggo, S. Robinson, M. Kunc, and Simon J. E. Taylor. Strengthening the reporting of empirical simulation studies: Introducing the STRESS guidelines. *J. Simulation*, 2019.

[22] R. Sargent. Verification and validation of simulation models. *Journal of Simulation*, 2013.

[23] Samuel Schmidgall, Rojin Ziaei, Carl Harris, Ji Woong Kim, Eduardo Pontes Reis, Jeffrey Jopling, and Michael Moor. AgentClinic: a multimodal benchmark for tool-using clinical AI agents. *npj Digital Medicine*, 9(1):499, 2026. arXiv:2405.07960.

[24] John F. Shortle, James M. Thompson, Donald Gross, and Carl M. Harris. *Fundamentals of Queueing Theory*. Wiley Series in Probability and Statistics. Wiley, 5 edition, 2018.

[25] Karandeep Singh, N. Shah, and A. Vickers. Assessing the net benefit of machine learning models in the presence of resource constraints. *J. Am. Medical Informatics Assoc.*, 2023.

[26] Ming Tai-Seale, Ellis C. Dillon, Yan Yang, Robert Nordgren, Ruth L. Steinberg, Teresa Nauenberg, Tim C. Lee, Amy Meehan, Jinnan Li, Albert Solomon Chan, and Dominick L. Frosch. Physicians' Well-Being Linked To In-Basket Messages Generated By Algorithms In Electronic Health Records. *Health Affairs*, 38(7):1073–1078, 2019.

[27] Yee Lam Elim Thompson, Gary M. Levine, Weijie Chen, Berkman Sahiner, Qin Li, Nicholas Petrick, Jana G. Delfino, Miguel A. Lago, Qian Cao, and Frank W. Samuelson. Applying queueing theory to evaluate wait-time-savings of triage algorithms. *Queueing Systems*, 108(3-4):579–610, 2024.

[28] Korosh Vatanparvar, Ashutosh Joshi, Maria Xenochristou, Mohammad Abuzar Hashemi, Prasad Kasu, Deepak Bansal, Daniel Lopez-Martinez, Anchal Nema, Ramya Ganesan, Will Kimbrough, Alex Woody, Yadunandana Rao, Dilek Hakkani-Tur, and Wilko Schulz-Mahlendorf. PatientAgentBench: A Benchmark Framework for Evaluating Patient-Facing Health AI Agents. *arXiv preprint arXiv:2607.25485*, 2026.

[29] Zi-Yao Wang and S. Rachev. Operating Imperfect AI: Reliability Drift and Human Congestion, 2026. arXiv preprint arXiv:2601.22295.

[30] Andrew Wong, E. Ötleş, J. Donnelly, A. Krumm, J. McCullough, Olivia DeTroyer-Cooley, J. Pestrue, Marie Phillips, Judy Konye, C. Penoza, M. Ghous, and Karandeep Singh. External Validation of a Widely Implemented Proprietary Sepsis Prediction Model in Hospitalized Patients. *JAMA Internal Medicine*, 2021.

[31] M. Wornow, Elsie Gyang Ross, A. Callahan, and N. Shah. APLUS: A Python library for usefulness simulations of machine learning models in healthcare. *Journal of Biomedical Informatics*, 2023.

[32] Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2406.12045.

[33] Shaoxin Zhong, Yuchen Su, and Michael Witbrock. Separating Diagnosis from Control: Auditable Policy Adaptation in Agent-Based Simulations with LLM-Based Diagnostics. *arXiv preprint arXiv:2603.22904*, 2026.