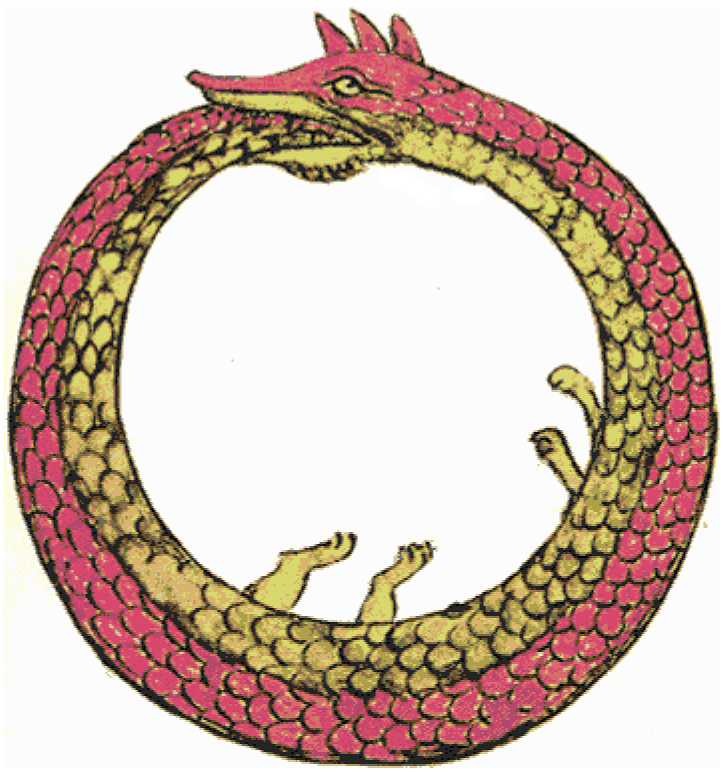
۱. جهان‌شمولی (Universality)

جهان‌شمولی در محاسبات به این معناست که یک ماشین محاسبه‌گر (مانند ماشین تورینگ) می‌تواند هر مسئله‌ای را که با الگوریتم قابل حل است، حل کند؛ به شرط آنکه به آن ماشین برنامه و ورودی مناسب داده شود.

۲. ارجاع به خود (Self-reference)



ارجاع به خود در محاسبات زمانی رخ می‌دهد که یک برنامه یا الگوریتم بتواند به خودش دسترسی داشته باشد یا خودش را تحلیل کند. مانند وقتی که یک تابع خودش را به صورت بازگشتی صدا میزند یا یک حلقه به برچسپ خودش بازگشت می کند. در زمینه محاسبات، ارجاع به خود به چالش‌های جالبی مانند مسئله توقف (Halting Problem) منجر شده است. مسئله توقف بیان می‌کند که هیچ الگوریتمی وجود ندارد که بتواند برای هر برنامه و هر ورودی تعیین کند که آیا آن برنامه متوقف خواهد شد یا نه.

۳. تفصیر کردن

اکثر مدل‌های محاسباتی که به اندازه کافی قدرتمند و غنی باشند (مانند ماشین تورینگ یا زبان‌های برنامه‌نویسی کامل تورینگ)، توانایی «شبیه‌سازی خودشان» را دارند. اما برای شبیه‌سازی خود یا دیگر سیستم‌ها، محدودیت‌هایی از نظر زمان و اندازه (فضا) وجود دارد که وابسته به مدل خاص محاسباتی است.

۱. شبیه‌سازی یک زبان توسط خودش

اگر یک زبان بتواند خود را شبیه‌سازی کند (مانند مفسر نوشته‌شده در همان زبان)، اندازه و زمان شبیه‌سازی به پیچیدگی زبان و میزان انتزاع آن بستگی دارد. در مورد زبان NAND-CIRC این پیچدگی برا است با

$$O(s) = s * \log(s)$$

برای محاسبه‌ی حد بالای فضا و زمان یک برنامه به دو پارامتر نیاز داریم. پیچیدگی و طول ورودی.
از فرمول بالا میتوانیم نتیجه بگیریم که اگر یک برنامه‌ی پایتون Tn دستور داشته باشد و طول ورودی ای برنامه حداکثر n خط باشد، برنامه‌ی NAND-CIRC با طول

$$O(T(n) \log T(n))$$

وجود دارد که با برنامه‌ی پایتون برابر است.

تز چرچ-تورینگ (Church-Turing thesis)

هر فرآیند محاسباتی که بتوان آن را به‌طور مکانیکی تعریف کرد، می‌تواند توسط یک ماشین تورینگ شبیه‌سازی شود. ازین تز می توان نتیجه گرفت که هر زبان برنامه نویسی تورینگ-کاملپت می‌توانند یک زبان برنامه نویسی تورینگ-کامپلت دیگر را شبیه سازی کند، و بحث فقط سر منابع مورد نیاز برای این کار است.

تز گسترش‌یافته چرچ-تورینگ فیزیکی (PECTT)

هر فرآیند فیزیکی که بتواند به‌عنوان یک محاسبه مورد استفاده قرار گیرد، می‌تواند توسط یک ماشین تورینگ کلاسیک با **منابع فیزیکی محدود** مدل شود.

> **“Physical Extended Church-Turing Thesis” (PECTT):** *If a function $F:\{0,1\}^n \rightarrow \{0,1\}^m$ can be computed in the physical world using s amount of “physical resources” then it can be computed by a Boolean circuit program of roughly s gates.*

منابع فیزیکی (Physical Resources):

در دنیای واقعی، منابع فیزیکی شامل انرژی، زمان، فضا (حافظه) و دقت اندازه‌گیری می‌شوند.
برای محاسبات کلاسیک، این منابع معمولاً در محدوده پلی‌نومی (polynomial) با اندازه ورودی افزایش می‌یابند.

گیت‌های منطقی (Logical Gates):

در محاسبات دیجیتال، هر محاسبه توسط مجموعه‌ای از گیت‌های منطقی مانند NAND انجام می‌شود.
تز گسترش‌یافته چرچ-تورینگ معمولاً فرض می‌کند که تعداد گیت‌های مورد نیاز و پیچیدگی زمانی محاسبه از مرتبه O fn برای توابع قابل محاسبه باشد، که در مدل‌های کلاسیک معمولاً این f پلی‌نومی است.

نگاه ریاضی به این قضیه:

به عبارت دیگر، می‌توان تز گسترش‌یافته چرچ-تورینگ فیزیکی $(PECTT)$ را این‌گونه بیان کرد که هر تابعی که توسط دستگاهی محاسبه شود که حجمی به اندازه (V) از فضا را اشغال می‌کند و برای انجام محاسبه به زمانی معادل (t) نیاز دارد، باید توسط یک مدار بولی با تعداد گیت‌هایی برابر با $(p(V,t))$ که چندجمله‌ای از (V) و (t) است، قابل محاسبه باشد.

شکل دقیق تابع $(p(V,t))$ به طور جهانی مورد توافق نیست، اما عموماً پذیرفته شده است که اگر تابعی مانند $(f:\{0,1\}^n \rightarrow \{0,1\})$ به‌طور نمایی سخت باشد (به این معنا که هیچ برنامه‌ای مبتنی بر NAND-CIRC با کمتر از $(2^{n/2})$ خط برای آن وجود نداشته باشد)، آنگاه اثبات وجود یک دستگاه فیزیکی که بتواند این تابع را در دنیای واقعی برای ورودی‌هایی با طول متوسط (مثلاً $(n=500)$) محاسبه کند، نقضی بر تز گسترش‌یافته چرچ-تورینگ فیزیکی خواهد بود. (گیج نشید، یعنی ثابت می کنیم مدارش اینقد ساختنش سخته (نمایی

سخت)، اما داریم فیزیکی انقد اسون تر (پولی‌پونمی) حلش می کنیم. پس داریم تز گسترش‌یافته چرچ-تورینگ فیزیکی رو نقض می کنیم.

تلاش برای به چالش کشیدن تز گسترش‌یافته چرچ-تورینگ فیزیکی

1. Spaghetti Sort (مرتب‌سازی اسپاگتی):

پیشنهادی برای یک "کامپیوتر مکانیکی" است که ادعا می‌کند می‌توان با استفاده از رشته‌های اسپاگتی به اندازه‌های مشخص مرتب‌سازی را سریع‌تر از محدودیت $(\Omega(n \log n))$ انجام داد. ایده این است که رشته‌ها را با طول مقادیر مرتب‌سازی بریده و روی سطح صاف قرار دهیم تا مرتب شوند. با این حال، این روش به دلیل محدودیت‌های فیزیکی و نظری، نمی‌تواند تز چرچ-تورینگ فیزیکی را نقض کند.

2. Soap Bubbles (حباب‌های صابون):

برای حل مسئله سختی مانند درخت اشتاینر اقلیدسی، پیشنهاد شده که از حباب‌های صابون برای بهینه‌سازی طول خطوط بین نقاط استفاده شود. (من نفهمیدم چطوری کار می کند) اما هرچند این روش در تعداد کم نقاط موفق است، اما با افزایش تعداد نقاط، به دلیل گیر افتادن در مینیمم‌های محلی، نتایج بهینه ارائه نمی‌دهد و نمی‌تواند تز را نقض کند.

3. DNA Computing (محاسبات با DNA):

استفاده از DNA برای حل مسائل محاسباتی سخت پیشنهاد شده است. DNA توانایی ذخیره اطلاعات با تراکم بالا و انجام محاسبات موازی را دارد، اما این چالش مهمی برای PECTT نیست، چرا که همچنان به منابع فیزیکی محدود وابسته است. همینطور ذخیره سازی دیجیتال ما نیز هر سال پیشرفته تر میشود و بعید نیست روزی از DNA هم فشرده تر شود.

4. Continuous/Real Computers (کامپیوترهای پیوسته):

اعداد حقیقی پیوسته اند، بیایید فرض کنیم که می‌توان آن ها را به صورت آنالوگ ذخیره و پردازش کرد. پیشنهاد می‌شود که دستگاه‌های آنالوگ می‌توانند از مقادیر واقعی استفاده کرده و قوی‌تر از مدل‌های گسسته باشند. با

این حال، افزایش دقت در اندازه‌گیری مقادیر پیوسته نیاز به منابع بیشتری دارد، و بنابراین نمی‌توان تز را نقض کرد. به علاوه هیچوقت نمی توان به نهایت دقت حقیقی در اندازه گیری مقادیر پویسته رسید.

5. Relativity Computer and Time Travel (کامپیوترهای نسبیتی و سفر در زمان):

ایده‌هایی مانند استفاده از نظریه نسبیت برای تسریع محاسبات یا استفاده از سفر در زمان (CTC) برای انجام محاسبات نامحدود پیشنهاد شده‌اند. هرچند این ایده‌ها جذاب هستند، اما نیاز به انرژی یا شرایط غیرممکن دارند و نمی‌توانند تز را نقض کنند. برای درک این مساله فرض کنید شما state یک تابع را ذخیره کرده و مرتبا با خود به گذشته ببرید. طبیعی است که سرعت پردازش شما بی‌نهایت سریعتر میشود.

6. Humans (انسان‌ها):

مغز انسان به‌عنوان یک سیستم محاسباتی مطرح شده است، اما توانایی‌های محاسباتی مغز (حدود (10^{14}) گیت در هر ثانیه) می‌توانند توسط مدارهای بولی شبیه‌سازی شوند. هرچند یافتن این شبیه‌سازی ممکن است دشوار باشد، اما ثابت نمیکند که توانایی انسان‌ها فراتر از ماشین‌های محاسباتی است. این تخمین های زده شده همه حد بالا اند، و در عمل وقتی در آزمایشگاه آن را انجام میدهیم، مثلا مغز یک موش را با نورون میسازیم، تعدادش بسیار کمتر میشود.

7. Quantum Computation (محاسبات کوانتومی):

محاسبات کوانتومی جدی‌ترین چالش برای PECTT است. ایده این است که سیستم‌های کوانتومی می‌توانند محاسباتی را انجام دهند که در مدل‌های کلاسیک غیرعملی هستند. در حال حاضر کامپیوترهای کوانتومی مقیاس‌پذیر ساخته نشده‌اند، اما این موضوع ممکن است نیاز به بازنگری تز را ایجاد کند. با این حال، اکثر مفاهیم اساسی در مدل محاسبات کلاسیک و کوانتومی مشترک هستند.

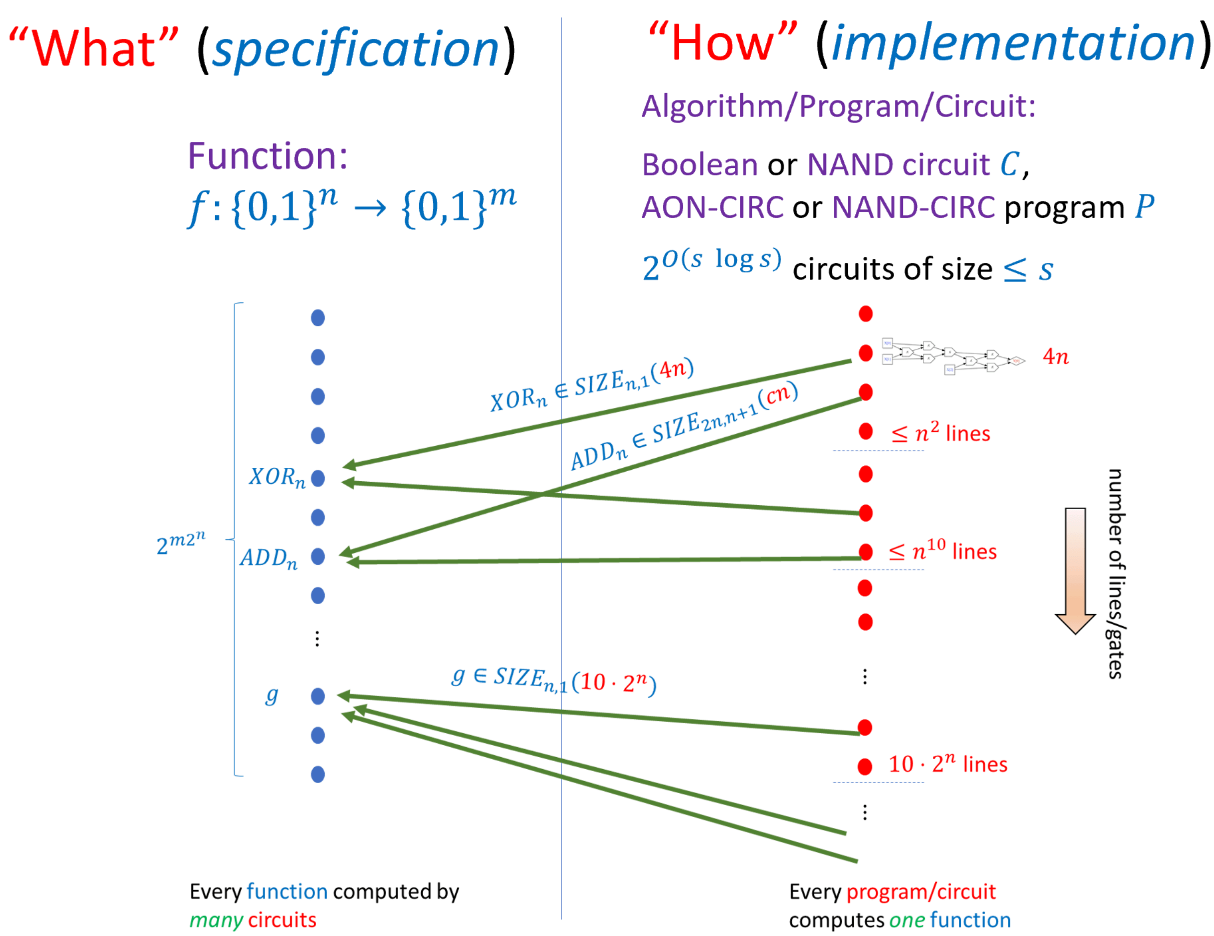
اساس رمزنگاری

در رمزنگاری کاربردی، اغلب با عباراتی مانند "سیستم رمزنگاری (X) امنیت ۱۲۸ بیتی ارائه می‌دهد" مواجه می‌شویم. این عبارت در واقع به این معناست که:

الف) فرض می‌شود هیچ مداری بولی (یا به‌طور معادل، برنامه‌ای مبتنی بر NAND-CIRC) با اندازه‌ای بسیار کمتر از (2^{128}) نمی‌تواند (X) را بشکند.
ب) فرض می‌کنیم هیچ مکانیسم فیزیکی دیگری نمی‌تواند بهتر عمل کند و بنابراین شکستن (X) به حدود (2^{128}) "منابع" نیاز دارد.

ما می‌گوییم "فرض می‌شود" و نه "اثبات شده"، زیرا در حالی که می‌توان بیان کرد که شکستن این سیستم با مدارهای دارای (s)-گیت به‌عنوان یک حدس ریاضی دقیق ممکن نیست، در حال حاضر نمی‌توانیم چنین ادعایی را برای هیچ سیستم رمزنگاری غیرساده اثبات کنیم.

عکس مهم



خلاصه

محاسبه یک تابع

هر تابع $(f : \{0,1\}^n \to \{0,1\}^m)$ را می‌توان با استفاده از (s) عملیات اساسی محاسبه کرد. نوع این عملیات (AND/OR/NOT یا NAND) تفاوت زیادی ایجاد نمی‌کند. این محاسبه را می‌توان با مدار یا برنامه‌ای خطی توصیف کرد.

کلاس محاسباتی

مجموعه $(SIZE_{n,m}(s))$ شامل توابعی است که با مدارهای NAND با حداکثر (s) گیت قابل محاسبه هستند. این مجموعه همچنین برابر با توابعی است که با برنامه‌های $(NAND-CIRC)$ حداکثر (s)-خطی قابل محاسبه هستند یا مدارهای بولی با حداکثر (s) گیت (AND/OR/NOT).

محدودیت‌های محاسباتی

هر تابع $(f : \{0,1\}^n \to \{0,1\}^m)$ می‌تواند با مدارهایی با حداکثر $(O(m \cdot 2^n/n))$ گیت محاسبه شود. برخی توابع حداقل به همین تعداد گیت نیاز دارند. کلاس $(SIZE_{n,m}(s))$ شامل توابعی است که می‌توانند با حداکثر (s) گیت محاسبه شوند.

مدارهای جهان‌شمول

مدار یا برنامه (P) را می‌توان به‌صورت یک رشته توصیف کرد. برای هر (s)، مدار جهانی (U_s) وجود دارد که می‌تواند برنامه‌هایی به طول (s) را با استفاده از توصیف رشته‌ای آن‌ها اجرا کند. این توصیف به ما امکان می‌دهد تعداد مدارهایی با حداکثر (s) گیت را بشماریم و نشان دهیم که برخی توابع با مدارهای کوچک‌تر از اندازه نمایی قابل محاسبه نیستند.

تز گسترش‌یافته چرچ-تورینگ فیزیکی

اگر تابع (f) با مداری دارای (s) گیت محاسبه شود، می‌توان یک دستگاه فیزیکی با (s) قطعه ساخت که (f) را محاسبه کند. تز گسترش‌یافته چرچ-تورینگ فیزیکی (PECTT) بیان می‌کند که عکس این موضوع نیز صحیح است: هر دستگاه فیزیکی برای محاسبه (f)، به حدود (s) منابع فیزیکی نیاز دارد. چالش اصلی در برابر PECTT محاسبات کوانتومی است.