Теорія складності

2. Теорія обчислюваності

2.6 Поняття класу складності

При побудові класифікації обчислювальних задач одним з головних елементів є поняття класу складності. Класи складності також використовують при розширенні класифікації на всі задачі: розв'язні та нерозв'язні.

Означення 2.1. *Класом складності* називають довільну множину мов.

***Зауваження*.** Без втрати загальності розглядають тільки мови над алфавітом $\{0,1\}$, тобто будь-який клас складності є підмножиною булеану $2^{\{0,1\}^*}$, а будь яка формальна мова над алфавітом $\{0,1\}$ є елементом цього булеану.

Це формальне означення класу складності мов в більш загальному розумінні. Наприклад, в такому розумінні класом складності є множина всіх мов над алфавітом $\{0,1\}$, які містять слово 010. Але треба розуміти, що не кожна множина мов є корисною для створення класифікації мов та відповідних задач за складністю. Створення класифікації задач за складністю передбачає визначення класів складності за певними характеристиками ресурсів, які є необхідними для розпізнавання чи вирішення мов з цих класів складності. Також корисними вважаються певна ієрархія введених класів складності за відношенням включення як множин та їхня замкненість відносно певних операцій над мовами. Саме замкненість відносно деяких операцій відіграє важливу роль при розрізненні різних класів складності. Також бажано, щоб клас складності мав практичну цінність, тобто, певні цікаві особливості або характеризував важливі практичні задачі, а також не залежав від обраної схеми кодування задач та навіть від обґрунтованих змін обчислювальної моделі чи алгоритмічної системи.

Зважаючи на наведене зауваження зазвичай використовують менш формальне і більш вузьке означення класу складності.

Означення 2.2. *Класом складності* називають множину мов, які можна розпізнати за певних обмежень на ресурси, що при цьому використовують.

Зауваження. Повністю формалізувати всі можливі обмеження на ресурси, які використовуються при розпізнаванні мов, є неможливим. Тому це означення можна сприймати як рекомендацію визначати класи складності саме по відношенню до кількості використовуваних ресурсів. Зазвичай нові класи складності з'являються з появою певних задач, які мають дещо інші характеристики в межах деякого наявного класу складності, з метою виділити ці задачі в окремий клас складності. Або, навпаки, якщо вводяться додаткові обмеження на використовувані ресурси, що здаються природними для створення класифікації.

Зазвичай при використанні машини Тюрінга в якості моделі обчислень клас складності визначають різновидом машини Тюрінга, яка використовується, та обмеженнями зверху на кількість тактів роботи та/або кількість використовуваних комірок стрічки, але обмеження і ресурси можуть бути довільними.

Нагадаємо, що існує певний взаємозв'язок між масовими задачами розпізнавання, всюди визначеними функціями виду $f\colon \{0,1\}^* \to \{0,1\}$ та мовами над алфавітом $\{0,1\}$. Між всюди визначеними функціями виду $f\colon \{0,1\}^* \to \{0,1\}$ та мовами над алфавітом $\{0,1\}$ існує взаємно однозначна відповідність, яка використовує характеристичну функцію мови над алфавітом $\{0,1\}$ як множини. Тому властивості формальної мови належності певному класу складності або іншим наслідкам будемо також асоціювати з відповідною функцією.

А масовій задачі розпізнавання Π відповідає мова $L[\Pi, e] \subseteq \{0,1\}^*$, яка є результатом застосування певної "розумної" схеми кодування до множини індивідуальних задач масової задачі Π, правильною розв'язком яких є стверднa відповідь. Зауважимо, що не завжди одній масовій задач відповідає одна формальна мова над алфавітом $\{0,1\}$, оскільки "розумних" схем кодування може бути декілька. Тому властивості належності певному класу складності або іншим наслідкам *кожної* формальної мови, утвореної за допомогою "розумної" схеми кодування, також асоціюють з відповідною масовою задачею розпізнавання. Бажано, щоб класи складності не залежали від схеми кодування задач, тобто, щоб всі мови, які є результатом застосування "розумних" схем кодування до множини індивідуальних задач деякої масової задачі завжди належали одному класу складності. В такому випадку властивості цього класу складності можна асоціювати з цією масовою задачею без додаткових обмежень.

Оскільки класом складності є множина мов, то до класів складності як до множини об'єктів є застосовними всі теоретико-множинні операції та відношення: об'єднання, перетин, різниця, доповнення, включення та інші.

З точку зору об'єктів — символів алфавіту, клас складності це множина множин впорядкованих множин символів алфавіту, символів 0 та 1. Теоретико-множинні операції та відношення можуть застосовуватися на кожному рівні абстракції, тому потрібно уважно слідкувати, до яких об'єктів застосовуються операції. Наприклад, операцію перетину можна застосувати до класів складності, формальних мов і навіть слів, якщо визначити результатом множину спільних символів двох слів, а ще можливі операції з символами алфавіту. Це дозволяє визначити додаткові операції над класами складності.

Означення 2.3. Нехай C_1 і C_2 є довільними класами складності.

Комплексним перетином (або *булевим перетином*) класів складності C_1 і C_2 називають бінарну операцію, результатом якої є клас складності $\{\, L_1 \cap L_2 \mid L_1 \in C_1, L_2 \in C_2 \,\}$, і яку позначають як $C_1 \wedge C_2$.

Комплексним об'єднанням (або *булевим об'єднанням*) класів складності C_1 і C_2 називають бінарну операцію, результатом якої є клас складності $\{\, L_1 \cup L_2 \mid L_1 \in C_1, L_2 \in C_2 \,\}$, і яку позначають як $C_1 \vee C_2$.

Також існує операція з класами складності, яку досить часто використовують, схожа на доповнення як множини, але доповнення застосовують не до множини мов, а до самих мов.

Означення 2.4. *Комплексним доповненням* (або *булевим доповненням*) довільного класу складності C_1 називають унарну операцію, результатом якої є клас складності $\{\, L_1 \subseteq \{\, 0, 1 \,\}^* \mid \overline{L_1} \in C_1 \,\}$, і яку позначають як coC_1.

Зауваження. Для довільного класу складності існує відповідний co клас складності, але в деяких випадках він збігається з іншим відомим класом складності або не має цікавих особливостей, тому відповідна назва не є загальновживаною. А в деяких випадках відповідний *co* клас складності навіть може мати альтернативне визначення, яке не використовує операцію комплексного доповнення.

Приклад 2.1. Нехай мова $L_0 \subset \{\,0,1\,\}^*$ складається з усіх слів, які починаються з символу 0, а мова $L_1 \subset \{\,0,1\,\}^*$ складається з усіх слів, які починаються з символу 1.

Визначимо класи складності

$C_0 = \{L_2 \subset \{\,0,1\,\}^* \mid L_2$ містить тільки слова, які починаються з символу 0 або порожнє слово, але $L_2 \neq \varnothing$ і $L_2 \neq \{\varepsilon\}\}$,

$C_1 = \{L_2 \subset \{\,0,1\,\}^* \mid L_2$ містить тільки слова, які починаються з символу 1 або порожнє слово, але $L_2 \neq \varnothing$ і $L_2 \neq \{\varepsilon\}\}$,

$C_2 = \{\varnothing, \{\varepsilon\}\} \cup \{L_2 \subset \{\,0,1\,\}^* \mid L_2$ обов'язково містить принаймні одне слово, яке починається з символу 0, і принаймні одне слово, яке починається з символу $1\}$ і

$C_3 = \{\,\varnothing, \{\,0,1\,\}^*\,\}$.

Безпосередньо з означень класів складності C_0, C_1, C_2 і C_3 випливає, що

- $C_0 \cup C_1 \cup C_2 = 2^{\{\,0,1\,\}^*}$;
- $C_0 \cap C_1 = C_1 \cap C_2 = C_0 \cap C_2 = \varnothing$;
- $C_0 \wedge C_1 = \{\,\varnothing, \{\,\varepsilon\,\}\,\}$;
- $C_0 \wedge C_0 = C_0 \wedge C_2 = C_0 \cup \{\,\varnothing, \{\,\varepsilon\,\}\,\}$;
- $C_1 \wedge C_1 = C_1 \wedge C_2 = C_1 \cup \{\,\varnothing, \{\,\varepsilon\,\}\,\}$;
- $C_0 \cup C_1 = 2^{\{\,0,1\,\}^*} \setminus C_2$;
- $C_0 \cup C_2 = 2^{\{\,0,1\,\}^*} \setminus C_1$;
- $C_1 \cup C_2 = 2^{\{\,0,1\,\}^*} \setminus C_0$;
- $C_0 \vee C_1 = C_2 \setminus \{\,\varnothing, \{\,\varepsilon\,\}\,\}$;
- $C_0 \vee C_2 = (C_0 \cup C_2) \setminus \{\,\varnothing, \{\,\varepsilon\,\}\,\}$;
- $C_1 \vee C_2 = (C_1 \cup C_2) \setminus \{\,\varnothing, \{\,\varepsilon\,\}\,\}$;
- $coC_0 = (C_0 \setminus \{\,L_0\,\}) \vee \{\,L_1\,\}$;
- $coC_1 = (C_1 \setminus \{\,L_1\,\}) \vee \{\,L_0\,\}$;
- $coC_2 = 2^{\{\,0,1\,\}^*} \setminus ((\{\,L_0\,\} \vee (C_1 \cup \{\,\varnothing\,\})) \triangle (\{\,L_1\,\} \vee (C_0 \cup \{\,\varnothing\,\})))$;
- $\overline{C_3} = 2^{\{\,0,1\,\}^*} \setminus \{\,\varnothing, \{\,0,1\,\}^*\,\}$;

- $coC_3 = C_3$;
- для довільного класу складності $C_4 \in 2^{\{0,1\}^*}$ $C_3 \wedge C_4 = C_4 \cup \{\varnothing\}$;
- для довільного класу складності $C_4 \in 2^{\{0,1\}^*}$ $C_3 \vee C_4 = C_4 \cup \{\{0,1\}^*\}$.

Зауважимо, що згідно з позначеннями математичні об'єкти ε, $\{\varepsilon\}$, $\varnothing$ та $\{\varnothing\}$ є абсолютно різними: ε позначає порожнє слово; $\{\varepsilon\}$ — мову, яка складається з одного елементу, порожнього слова; $\varnothing$ — порожню мову, яка не містить жодного слова (також може позначати порожній клас складності, який не містить жодної мови); $\{\varnothing\}$ — клас складності, який складається з одного елементу, порожньої мови.

Для класів складності важливим є властивість замикання його мов відносно основних операцій над мовами. Повний список операцій над мовами, відносно яких клас складності є замкненим, можна використовувати навіть для розрізнення класів складності, якщо такі їх списки не збігаються.

Існує достатньо велика кількість різних класів складності, які використовуються для класифікації задач або відповідних формальних мов. Хоча треба зазначити, що для більшості класів досі немає точних результатів щодо відмінностей з іншими класами складності. В теорії складності використовується досить велика кількість припущень і ще більше існує питань без точної відповіді. Доведення рівності або нерівності класів є однією з найважчих задач класичної теорії складності.

Визначимо перші й одні з найбільших класів складності, з яких почнемо будувати ієрархію класів.

Означення 2.5. *Класом складності* ALL називають множину всіх мов над алфавітом $\{0,1\}$.

***Наслідок*.** Відповідно класом складності ALL є множина всіх функцій f виду $f\colon \{0,1\}^* \to \{0,1\}$. Також часто кажуть, що клас складності ALL є множиною всіх задач розпізнавання, але, згадуючи, що відповідність масових задач розпізнавання та формальних мов не є взаємно однозначною, то в загальному випадку може трапитися так, що не всім формальним мовам над алфавітом $\{0,1\}$ відповідає хоча б одна задача розпізнавання. Все одно будемо вважати, що клас складності ALL є множиною всіх задач розпізнавання, щоб зберегти універсальність підходу для формальних мов, задач розпізнавання та функцій, за необхідності, поставивши у відповідність довільній формальній мові L_1 над алфавітом $\{\,0,1\,\}$ задачу розпізнавання — розпізнати слова мови L_1.

Означення 2.6. *Класом складності* RE (від англ. recursively enumerable) називають множину всіх рекурсивно зліченних за Тюрінгом мов над алфавітом $\{\,0,1\,\}$.

Означення 2.7. *Класом складності* R (від англ. recursive) називають множину всіх вирішуваних за Тюрінгом мов над алфавітом $\{\,0,1\,\}$.

Відповідно, клас складності RE є множиною задач розпізнавання, для яких за допомогою машини Тюрінга можна перевірити відповідь, якщо правильною є ствердна відповідь. А клас складності R є множиною задач розпізнавання, для яких існує машина Тюрінга, що розв'язує цю задачу, або множиною повністю визначених обчислювальних функцій з областю значень $\{\,0,1\,\}$.

Також виділяють множину мов, які не є рекурсивно зліченними і не є корекурсивно зліченними, тобто мови, в яких неможливо алгоритмічно правильно визначити за скінченний час для довільного слова чи належить воно цій мові, а також неможливо алгоритмічно правильно визначити за скінченний час для довільного слова, що воно не належить цій мові. Фактично така множина складається з усіх мов, які не увійшли хоча б до одного з класів $coRE$ та RE.

Означення 2.8. *Клас складності* $NRNC$ визначають як $NRNC = ALL \setminus (RE \cup coRE)$.

З властивостей вирішуваних за Тюрінгом мов та рекурсивно зліченних за Тюрінгом мов випливає твердження.

Наслідок. $R \subseteq RE \subseteq ALL$ і $R \subseteq coRE \subseteq ALL$.

Безпосереднім наслідком теореми Поста є твердження.

Наслідок. $R = RE \cap coRE$.

Також, використовуючи вже відомі результати, можемо визначити належність введених раніше мов (та відповідних задач) до визначених класів складності. Так, $A_{TM}, HALT, HALT^{\varepsilon} \in RE$, також $E_{TM}, EQ_{TM} \in coRE$, але при цьому всі ці мови не належать класу складності R. Мова INF_{TM} не належить жодному з класів RE та $coRE$, тобто $INF_{TM} \in NRNC$.

Наслідок. З існування рекурсивно зліченної мови, яка не є вирішуваною, та існування мови, яка не є рекурсивно зліченною та корекурсивно зліченною, випливає, що

- $R \neq RE$, $R \neq coRE$, $RE \neq coRE$;
- $NRNC \neq \varnothing$;
- $R \subset RE \subset ALL$, $R \subset coRE \subset ALL$.

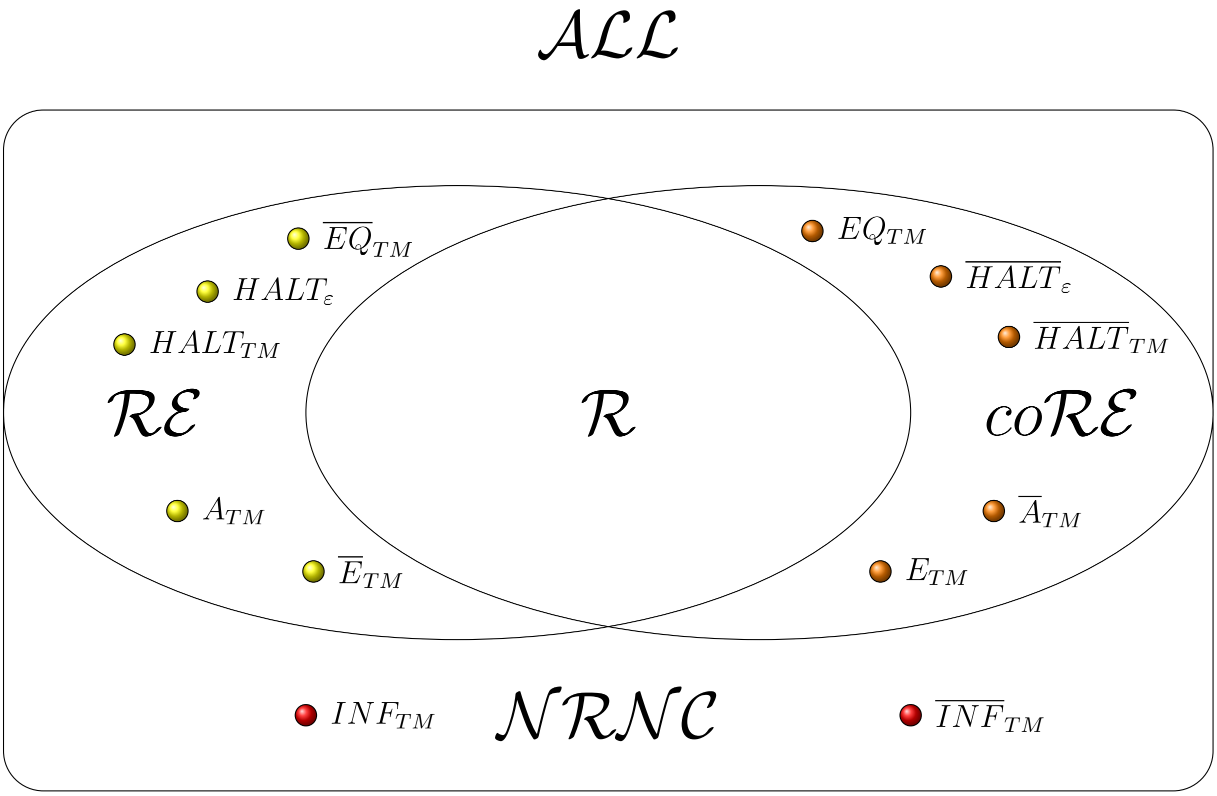

Рисунок 1.1 — Діаграма класів складності із задачами

Твердження 2.1. Клас складності R є замкненим відносно операцій об'єднання, перетину та конкатенації мов, а також відносно операції замикання Кліні та доповнення мови.

Клас складності RE є замкненим відносно операцій об'єднання, перетину та конкатенації мов, а також відносно операції замикання Кліні, але не є замкненим відносно операції доповнення мови.

Клас складності $NRNC$ є замкненим відносно операції доповнення мови, але не є замкненим відносно операцій об'єднання та перетину мов.

Клас складності ALL є замкненим відносно будь-якої операції над мовами.

$\triangleright$ Розглянемо довільну мову L_1 з класу складності $NRNC$. За означенням класу складності $NRNC$ мова L_1 не належить ні класу складності RE, ні класу складності $coRE$. Якщо припустити, що мова $\overline{L_1}$ не належить класу складності $NRNC$, то з цього випливає, що мова $\overline{L_1}$ належить класу складності RE або класу складності $coRE$ (або обом класам одночасно). Якщо $\overline{L_1} \in RE$, то це означає, що $L_1 \in coRE$, що є суперечністю. Якщо $\overline{L_1} \in coRE$, то це означає, що $L_1 \in RE$, що також є суперечністю. Отже, мова $\overline{L_1}$ не належить ні класу складності RE, ні класу складності $coRE$, а з цього випливає, що вона належить класу складності $NRNC$. Тобто, клас складності $NRNC$ є замкненим відносно операції доповнення мови.

Розглянемо довільну мову з класу складності $NRNC$, наприклад, мову INF_{TM}. Так як клас складності $NRNC$ є замкненим відносно операції доповнення мови, то з цього випливає, що мова $\overline{INF_{TM}}$ також належить класу складності $NRNC$. Перетин мов INF_{TM} та $\overline{INF_{TM}}$ є порожньою множиною, а об'єднання цих мов дорівнює $\{0, 1\}$. Для порожньої множини та мови $\{0, 1\}$ можна побудувати вирішувачі (які будуть працювати точно один такт з довільним вхідним словом), а це означає, що порожня множина та мова $\{0, 1\}$ належать класу складності R, тобто не належать класу складності $NRNC$. Отже, клас складності $NRNC$ не є замкненим відносно операцій об'єднання та перетину мов. $\square$

З доведення попереднього твердження також можна зробити цікавий висновок, що відношення включення між мовами не завжди має вплив на складність розпізнавання чи вирішування цих мов. Так, якщо відомо, що $L_1 \subseteq L_2$ для деяких мов L_1 та L_2, і, якщо L_2 є невирішуваною мовою, то жодного висновку не можна зробити щодо вирішуваності мови L_1 (аналогічно, коли відомо, що L_1 є невирішуваною мовою). Дійсно, відомо, що мова INF_{TM} є невирішуваною, і навіть не рекурсивно зліченною, але $\varnothing \subset INF_{TM} \subset \{0, 1\}^*$, де мови $\varnothing$ та $\{0, 1\}$ є не просто вирішуваними, а відповідні вирішувачі працюють один такт для будь-якого вхідного слова. А відношення, яке б зберігало певну міру складності мов, було б дуже корисним для побудови класифікації задач, тобто, наприклад таке бінарне відношення $\mathcal{R}$, що, якщо відомо, що $L_1 \mathcal{R} L_2$ для деяких мов L_1 та L_2, то це б означало, що розпізнати мову L_2 аж ніяк не простіше ніж розпізнати мову L_1.